Molecular modeling of fracture in methane hydrates

by

Henrik Andersen Sveinsson

Thesis

for the degree of

Master of Science

Faculty of Mathematics and Natural Sciences
University of Oslo

April 2015

Abstract

Elastic and failure properties of methane hydrates are studied using molecular dynamics simulations. The TIP4P/Ice water model and the OPLS united atom methane model are employed in the study. Mechanical properties are reported, and a possible fracture initiation process is identified. On the nanosecond timescale, a pure sI methane hydrate is identified as brittle, and with a fracture toughness of ≈ 0.06 MPam$^{\frac{1}{2}}$. The initiation of cracks in the modeled systems is highly dependent on slow dissociation (melting) of the hydrate prior to rapid crack propagation. The melting occurs on the surface of initial and artificial flaws introduced in the hydrate. Furthermore, methane is immediately released upon fracture, while water molecules stick to the crack walls. This work provides some first steps into molecular dynamics studies of fracture in methane hydrates.

Acknowledgements

I met my supervisor, Professor Anders Malthe-Sørenssen, during my first semester of Bachelor's studies at the University of Oslo, and I was quickly taken in as part of a research group under his supervision. Later, Anders became my supervisor for the master's project, and I thank him for providing excellent feedback and advice through the work with this thesis.

While working on this project, I have had the pleasure of sharing an office with Ph.D. students Kjetil Thøgersen, Anders Hafreager and Jørgen Trømborg. I thank them for useful discussions, both related to this thesis and everything else.

I would also like to thank György Hantal and Alexandru Botan for helpful discussions on how to impose fracture in molecular dynamics.

Finally, I want to thank friends and family, and of course Mari, for enriching my life outside of physics.

Oslo, 15 April 2015
Henrik Andersen Sveinsson

Contents

Abstract **iii**

Acknowledgements **v**

1 Introduction **11**
1.1 The art of computer simulations 11
1.2 Methane hydrates 12
1.3 The ethics of hydrocarbon research 13
1.4 My contribution 14
1.5 Goals and outcomes 15
1.6 Structure of this thesis 16

I Background **17**

2 Methane hydrates **19**
2.1 History, occurrence and resource potential 19
2.2 Molecular structure 20
2.3 Mechanical properties from experiments 23
2.3.1 Typical experimental setup 24
2.3.2 Experimental results 26
2.4 Classical potential models of water 26
2.5 Molecular dynamics modeling of methane hydrates 28
2.6 Quantum mechanical calculations on methane hydrates 29
2.7 Molecular dynamics modeling of fracture 30
2.8 Research questions 30

3 Elasticity and failure **33**
3.1 Linear elasticity 33
3.1.1 Stress and strain 33
3.1.2 Hooke's law 34
3.1.3 Isotropic materials 36
3.1.4 Plane strain and plane stress 36
3.1.5 Elastic moduli 37
3.1.6 Elastic waves 37

3.2 Linear elastic fracture mechanics . . . 38
3.2.1 Brittle and ductile materials . . . 38
3.2.2 Modes of loading . . . 38
3.2.3 Failure criteria . . . 38
3.2.4 Stress intensity factors in anisotropic materials . . . 43
3.3 Stress concentrations around an elliptical hole . . . 43

4 Molecular dynamics 45
4.1 Introduction . . . 45
4.2 Potentials . . . 45
4.2.1 Non-bonding potentials . . . 46
4.2.2 Bonding potentials . . . 47
4.3 Time integration . . . 47
4.3.1 Velocity Verlet . . . 47
4.4 Simulation box . . . 48
4.5 Temperature . . . 48
4.5.1 Berendsen thermostat . . . 49
4.5.2 Nosé–Hoover thermostat . . . 50
4.6 Pressure and stress tensor . . . 51
4.6.1 Barostats . . . 52
4.7 Cutoffs and long-range corrections . . . 52
4.7.1 Ewald summation . . . 53
4.8 Rigid groups of particles . . . 53
4.9 Measurements . . . 54
4.9.1 Self-diffusion coefficient . . . 54
4.9.2 Viscosity . . . 54
4.9.3 Radial distribution function . . . 54

II Numerical models 57

5 A model for methane hydrates 59
5.1 Criteria for a sufficient model . . . 59
5.2 Water models . . . 60
5.2.1 SPC water model . . . 60
5.2.2 TIP4P water model . . . 60
5.3 Optimized Potentials for Liquid Simulations (OPLS) . . . 61
5.3.1 Potential model . . . 61
5.3.2 OPLS United-atom methane . . . 62
5.4 Combining particles of different species in a model . . . 63
5.5 TIP4P-ICE + OPLS-UA methane . . . 63
5.6 A system for studying fracture of methane hydrates . . . 63

6 Simulation and analysis tools 67
6.1 LAMMPS . . . 68
6.1.1 Input files . . . 68

6.2 Extending LAMMPS to cut elliptical prisms . . . 73
6.2.1 Usage . . . 73
6.3 Crack tracer . . . 74
6.3.1 Algorithm . . . 74
6.3.2 Implementation . . . 75
6.3.3 Choice of parameters . . . 75
6.4 Stress measurement . . . 78
6.5 Other tools . . . 78

7 Verification of models in LAMMPS 81
7.1 TIP4P/Ice . . . 81
7.1.1 Density of bulk water . . . 81
7.1.2 Diffusivity and shear viscosity . . . 82
7.2 United atom methane . . . 84
7.3 Stabilizing methane hydrates . . . 84
7.4 Numerical efficiency . . . 86
7.5 Atoms out of range during P^3M calculation: A time-consuming bug . . . 86

8 Modeled systems 89
8.1 Shear viscosity and diffusivity of the water model . . . 89
8.2 Measuring elastic properties with a constant strain rate . . . 91
8.2.1 Lennard-Jones crystal . . . 92
8.2.2 sI methane hydrate with TIP4P/ICE+UAM . . . 92
8.3 Early results on fracture and fracture toughness . . . 94
8.3.1 Simulation protocol . . . 94
8.3.2 Proof-of-concept simulations . . . 96
8.3.3 Arising questions . . . 102
8.4 Energy considerations – the thermodynamics of expanding the simulation box . . . 103
8.5 Large simulations of fracture . . . 106
8.5.1 Complete analysis of a single simulation . . . 106
8.6 A simple characterization of the fracture: The amount of free methane. . 110
8.6.1 Estimate by counting . . . 114
8.6.2 Results on free methane . . . 114
8.6.3 Diffusion in the crack space . . . 116
8.7 Increasing the temperature . . . 117
8.8 Investigating simulations with no system-spanning crack . . . 118
8.9 Aggregated results . . . 119
8.9.1 Effects of the system thickness . . . 120
8.9.2 Waiting time for fracture . . . 120
8.9.3 Crack area and crack tip speed . . . 121
8.9.4 Free methane . . . 123
8.9.5 Surface energy and fracture toughness . . . 123
8.9.6 Comparison with linear elastic fracture mechanics . . . 123

III Summary and conclusions **127**

9 Summary and conclusions **129**
9.1 Summary and conclusions 129
9.2 Discussion 130
9.3 Outlook 131
9.4 Ending remarks 132

IV Appendices **135**

A Input file templates **137**
A.1 Template format 137
A.2 Python implementation 139

B Details of using the Abel computing cluster **141**
B.1 Compiling lammps on the Abel computing cluster 141
B.2 Submitting jobs 141
B.3 Experienced problems when using the Abel computing cluster 142

Bibliography **144**

Chapter 1

Introduction

In this chapter, I start by introducing computer simulations and methane hydrates, and discuss the ethics of hydrocarbon research. Then I go on to listing the contributions of my work, the goals and outcomes of the project, and describe the structure of this thesis. The methane hydrate introduction in this chapter is brief, and a more thorough description is given in chapter 2.

1.1 The art of computer simulations

The abundance of cheap computing resources developing over the last few decades has resulted in extensive computational studies in all parts of science. These studies have aided the understanding of physical phenomena and the development of new technologies. An early example is the atomic bomb. An everyday example of the application of computer simulations is weather forecasting. Increasing access to high-performance computers has resulted in weather forecasts that are useful several days in advance. In the first example – the atomic bomb – computer simulations aided physical understanding. In weather forecasting, however, we already know the physics. The model incorporates known laws of nature to predict weather conditions forward in time. I will be doing the former – but on another topic.

A physical model is a mathematical formulation of a physical problem with the purpose of capturing some aspect of the behavior of that system. A simulation is a solution of the equations defining the model. For some simple models, the equations can be solved analytically, and, in that case, the solution can be thought of as a simulation. For many models, we are not able to solve the equations analytically, and numerical methods must be applied to simulate the model – this is what is usually referred to as a simulation.

Computer simulations are superior when it comes to having the full state of a system available in all the timescales and spatial scales of the simulation, since these are available by design. That means simulations can aid understanding and predicting the behavior of systems on scales that are not experimentally available. Conversely, a computer simulation is *always* limited in time and space since computing power is finite. First-principles studies can only be performed on small and short scales. To do simulations on large systems over long times, some details must be omitted, resulting

in approximated models. The quality of these approximations are crucial, and the profound implication is that it is hard to know whether a simulation captures the relevant physics.

The approximations of the models represent a substantial difference between simulations and physical experiments. At the same time, computer simulations and physical models exist in close interplay with experiments. For a simulation of a model to be valuable, it has to reproduce relevant features of the physical system. These features are usually found in experiments. When a model is sufficiently calibrated from experimental data, it is possible, with some uncertainty, to use that model to study features that were not available from the experiment. Then, new experiments can be proposed to shed light on the findings from the model simulations, and possibly increase confidence in that model.

Computer models are a new thing, in the sense that running simulations on electronic computers was impossible before such devices were available. On the other hand, computer models are an old thing since mathematical models could be formulated for computers long before electronic computers were even proposed. Indeed, iterative schemes like Newton's method for roots of a function and Euler's method for the integration of differential equations were developed by people who never lived to see electronic computers. The computations would be performed by a human, who would at the time be referred to as a computer.

In the intersection between computational science and theoretical physics, we find the field of computational physics: Mathematical models are developed to answer physics questions, and these models are studied using methods of computational science. On the other hand, computational physics is a close relative of experimental physics in the way that the computational physicist perform "numerical experiments" when running simulations. However, the simulations are fully decoupled from the sensory world – they are not real experiments. Thus, the computational physicist is a theoretical physicist using a computer. Still, experimental intuition remains important, as computer simulations resemble experiments, and the analysis methods are similar.

1.2 Methane hydrates

Methane hydrates are so-called clathrate compounds, meaning that they consist of host molecules forming a lattice that traps guest molecules. Methane hydrates are a special case of *gas hydrates*, or *clathrate hydrates* which they are also called. In methane hydrates, water molecules form cages that can host single methane molecules. At first sight, a piece of methane hydrate will resemble regular ice. However, if kept at regular temperatures and pressures, it will release methane gas, allowing it to be set on fire – fiery ice.

When multiphase pipelines were introduced in the petroleum industry, finding the necessary physical conditions for methane hydrate precipitation became interesting and necessary. Multiphase pipelines – pipelines that transport oil, natural gas and water at the same time – are at risk of providing conditions where methane hydrates can form a plug that will decrease or stop flow in the pipeline, and possibly damage equipment. The first observation of such plugs was done in the 1930's [29].

Over the last 10-20 years, rising prices on fossil fuels have started a revolution in research, exploration and recovery of unconventional hydrocarbon resources. Indeed, around 50 % of all natural gas and oil recovered in the United States in 2014 came from hydrocarbon-bearing shales [3].

Methane hydrates have not received nearly as much attention as shale gas, despite the estimated amounts of hydrocarbon residing in methane hydrates being significantly larger than those of shale hydrocarbons. That might be because methane hydrates seem harder to extract. Gas hydrate extraction is still in the piloting phase, and it is still not commercially viable.

In addition to the interest in methane hydrates as an energy resource, is also the fear that methane hydrates can – if released – contribute significantly to out-of-control global warming. Methane hydrates are stable within some range of temperatures and pressures, but if brought out of that range, they dissociate and release the methane. Methane is a powerful greenhouse gas, and the idea is that the released methane will contribute to even more warming, releasing more methane in a process spinning out of control: *The clathrate gun hypothesis* [42]. This hypothesis is set both as a possible explanation of past events of global warming (on geological timescales) and a risk for the future.

Understanding the dissociation mechanisms of methane hydrates is important both with respect to extraction and with respect to the possibly hazardous behavior of methane hydrates during global warming.

1.3 The ethics of hydrocarbon research

Methane hydrates can be studied with the sole purpose of understanding their basic behavior, offering no attention to how that understanding might be used. But the reality is that if the political climate doesn't change during the coming years, knowledge about new hydrocarbons can result in higher CO_2-emissions. Therefore, the ethics of petroleum research need to be addressed.

The Intergovernmental Panel on Climate Change (IPCC) concludes in its synthesis report (SYR) from 2014 [55] that:

> The IPCC is now 95 percent certain that humans are the main cause of current global warming. In addition, the SYR finds that the more human activities disrupt the climate, the greater the risks of severe, pervasive and irreversible impacts for people and ecosystems, and long-lasting changes in all components of the climate system.

This means that we cannot continue burning fossil fuels at the rates that we have been doing over the last decade. A question naturally arising from that, is whether it is responsible to perform research that can potentially increase CO_2-emissions by making it easier to extract fossil fuels.

Fossil fuel dependence must by definition end at some point, but fossil fuel extraction cannot stop tomorrow. That would have a great impact on people's lives and standard of living, and an abrupt discontinuation of fossil fuel extraction would ultimately lead to higher mortality rates.

Fossil fuels can be ranked by how much CO_2 they emit per Joule of energy produced. In that respect, it would for instance be better to research gas than coal. But gas is not good enough unless it is only a bridge towards a fossil-free world.

Technologies for carbon capture and storage (CCS) are under development, and if they succeed, the climate is no longer the limiting factor for using fossil fuels. Then the game is suddenly changed, and research that enables extraction of previously unviable resources.

The Norwegian National Committee for Research Ethics in Science and Technology (NENT) conducted in 2014 an assessment of the ethics of Norwegian petroleum research. Their main conclusion, as stated on their web page is [70]:

> It is indefensible from a research ethics perspective if petroleum research hinders processes of transition to sustainable energy and thus prevents achievement of UN climate goals which Norway has pledged to uphold

This is not really a conclusion, since the premise is still very much up for debate. Does petroleum research hinder the transition to sustainable energy? These questions are still open and under debate. Here, I conclude that there *are* ethical concerns regarding petroleum research, but it is not obvious or how to answer or resolve them in a best possible way for humankind.

1.4 My contribution

This thesis contains descriptions of previous works and original contributions.

I have developed new tools for analysis, which are described in chapter 6. It should be clear from the description what parts of that chapter are my descriptions of tools made by others, and what tools were developed by me.

Most of my work has been on the properties of methane hydrates in the TIP4P/ICE water + OPLS-UAM methane model (explained in detail in chapter 5). This model, like any molecular dynamics model of methane hydrates, is poorly investigated. Additionally, fracture of methane hydrates is poorly studied, and to my knowledge, this is the first study of fracture of methane hydrates using molecular dynamics.

Below follows a list of my scientific contributions in this thesis:

- Transport properties of the TIP4P/Ice water model at 300 K. (Diffusion coefficient and viscosity).
- Mechanical properties of methane hydrates in TIP4P/Ice+OPLS-UAM: Young's modulus, Poisson's ratio.
- Fracture toughness of TIP4P/Ice + OPLS-UAM under tensile strain.
- Brittleness of TIP4P/Ice+OPLS-UAM under rapid loading.
- Separation of two distinct stages of fracture propagation in TIP4P/ICE+OPLS+UAM.

1.5 Goals and outcomes

The goal for my master project is to study dissociation mechanisms for methane hydrates on the molecular scale. I will use molecular dynamics to study dissociation through crack propagation due to externally applied stress in pure, crystalline methane hydrates with artificial defects. The aim is to reproduce some simple mechanical and fracture properties of these systems, but not to reproduce or explain any particular experiments or phenomena. Additionally, some possibly fruitful paths for future research will be identified.

The project was initially outlined with the following sub-projects:

1. **Develop molecular dynamics models for gas hydrate modeling:** The student will apply standard molecular dynamics codes, such as LAMMPS, to study gas hydrates in the stable regime. In addition, we will develop our own codes with reactive potentials, and validate the model by comparison with standard codes in the stable regime.

2. **Apply model to study gas hydrate dissociation:** The student will use the validated models to address gas hydrate dissociation processes in scenarios relevant for production.

3. **Refine model to include sedimentary interactions:** Finally, the student will develop the model to include effects of sediments, such as by including effects of silicate glasses for which we have well-tested reactive potentials, to address the effect of the surrounding sedimentary minerals and confined pore spaces on gas hydrate dissociation processes.

During the work on the first subproject, we decided that time could be better spent studying fracture as a dissociation mechanism in methane hydrates, since this kind of study has been proposed but not yet conducted. So in reality, the project has consisted of the following sub-projects:

1. **Develop molecular dynamics models for gas hydrate modeling:** The student will apply standard molecular dynamics codes, such as LAMMPS, to study gas hydrates in the stable regime.

2. **Apply model to find a protocol to study fracture of methane hydrates:** Use the validated model to find a way to impose and analyze fracture in the methane hydrate model.

3. **Refine the protocol for studying fracture, and characterize fracture in methane hydrates:** Use the developed protocol to run simulations and study dissociation and mechanisms for fracture initiation and propagation in methane hydrates.

Note that the project no longer contains the development of our own code. During the work on the first subproject, LAMMPS was found sufficiently flexible for developing our own potentials within its framework. This has not been done in this work because the fracture study was prioritized, not leaving time for the development of our own reactive potentials. But reactive potentials will be a priority in future work.

1.6 Structure of this thesis

In addition to being a scientific record, this thesis is also a documentation of my work during the last year. If I have been struggling with a topic and spent a lot of time figuring it out, this will be reflected in the thesis. These parts will be more elaborate than other parts of the thesis, regardless the scientific value of their contents. An example is the section "proof-of-concept runs" in the chapter about modeled systems. This section represents a lot of work to find the right parameters to go into the model, but the results have little impact beyond the validation of the model and its applicability to the problems I work with.

This thesis is arranged in four parts. Part I contains background chapters. It starts with chapter 2 on methane hydrates, which sums up relevant aspects of the state of the science of methane hydrates, both experimentally and theoretically, and ends with questions that my research aims to answer. Then comes two theory chapters: one on elasticity and failure, and one on molecular dynamics. I have sought to explain theory needed to for understanding the physics of a molecular dynamics simulation. That means implementation details that are unimportant to the actual outcome of my simulations are not discussed. Some topics are explained in detail, while others are too technical, requiring too many details to be interesting in a master thesis. In these cases, I will refer to research papers that can supplement my presentation. Part II is the main part of this thesis. In chapter 5 I introduce interaction potentials and choose which ones to use to describe methane hydrates. In addition, I describe the system that I will study (Initial particle positions, thermodynamic conditions and boundary conditions). Chapters 6 and 7 describes the numerical tools I utilize and the tools I have developed, along with a brief verification of the numerics. Chapter 8 contains the main work and results. This chapter is mostly organized chronologically after when I did what, and it can be seen as a description of the research process that I have been going through after I got the simulation tools to work properly. Some tools were developed during the research process, and these tools are described in detail in chapter 6, but it should be clear from the discussion in 8 when they were actually introduced. Part III contains only one chapter, chapter 9, where I summarize my results, conclude based on those results, and propose possible topics for future work. Part IV contains the appendices: Contents that have value but would hamper the flow of the document if they were presented in the main text.

Part I

Background

Chapter 2

Methane hydrates

Gas hydrates are clathrate compounds, which means they consist of host molecules forming a lattice that traps guest molecules. Special to clathrate *hydrates*, which are clathrates where water form the lattice, is that the structure is stabilized by the guests, and would collapse into regular ice or liquid water without guests present. Methane hydrates are among the more prevalent clathrate hydrates, and are formed by water molecules providing cages that host methane molecules. The most common cage structures are comprised of pentagonal and hexagonal faces forming regular cage structures. These structures can be described as replications of relatively simple unit cells. This work focus on methane hydrates, but since other clathrate hydrates have also been researched, and are relevant, they will be mentioned.

2.1 History, occurrence and resource potential

Gas hydrates were probably discovered as early as in the late 1700's by Sir Joseph Priestly, and the definitive confirmation of their existence was done by Sir Humphry Davy in 1810 [31]. Early studies focused on identifying possible guest molecules in gas hydrates, but when methane-containing hydrates were shown to form in pipelines in 1934 [29], *methane* hydrates got more attention. However, the discovery of methane hydrates occurring in natural deposits was not made until the 1960's [45]. Methane hydrates are now known to occur in mud, sand, permafrost and on the sea floor around the world, within what is called the *gas hydrate stability zone* (GHSZ). The GHSZ is characterized by low temperatures and high pressures compared to the stability zone of pure water ice. Figures 2.1 and 2.2 illustrate the stability zones in permafrost and under the sea floor, respectively. Special to the sea-floor conditions is that the deeper the water, the deeper the stability zone beneath the sea floor, since sea floor temperature is almost constant. The stability zone for methane hydrate starts at around 300 m in seawater and 100 m in permafrost [31]. In order for methane hydrates to actually occur, methane is needed. The methane is supplied either from biogenic or thermogenic methane sources. The geological setting of the hydrate formation is illustrated in figure 2.4 According to the World Ocean Review [13], methane hydrates are mainly found on the continental slopes and in arctic permafrost. Since methane hydrates reserves were discovered, estimates of total reserves have mainly been done for hydrates in marine

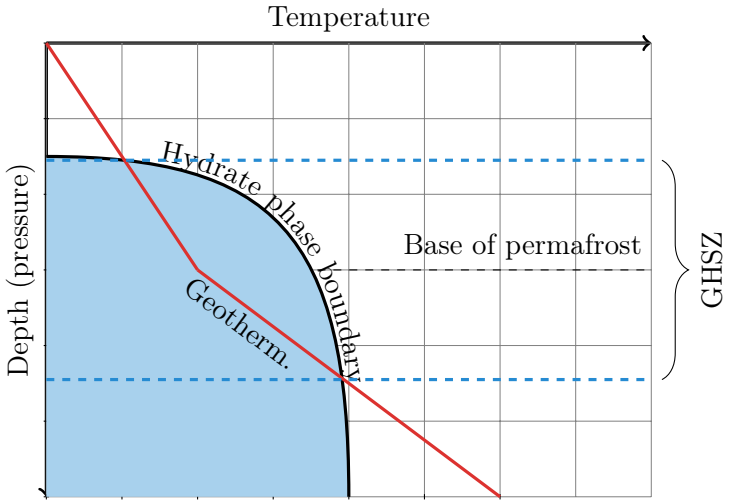


Figure 2.1: Gas hydrate stability conditions in permafrost (blue filled area). The red line shows the temperature as a function of deptth. In permafrost, the temperature is always increasing with increasing depth. This leaves a limited depth zone where the temperature–pressure configuration id within the gas hydrate stability conditions. This is the gas hydrate stability zone (GHSZ).

sediments. The estimates have varied greatly, but the overall tendency has been a decline in the estimated amounts. Early estimates were as high as 10 000 Gt of carbon, but it has later been argued that the an amount of 500–2500 Gt is more reasonable [50]. A fairly recent study by Wallmann *et al.* [75] estimates the global reserves of carbon in methane hydrates in marine sediments to more than 455 Gt. This estimate comes with a geographical distribution of the reserves, which is shown in figure 2.3. This estimate is much smaller than the early estimates, but it is still large compared to the *proven* natural gas reserves in the world, which is roughly 120 Gt (2013, [17]) of carbon. It must further be assumed that only a fraction of these resources are recoverable, but these numbers still leave the possibility of methane hydrates playing an important role in the world energy mix. Some countries, like Japan and India, are very coal-dependent and seek to diversify their energy mix. If methane hydrates turn out to be viable, they can be of great economic importance to such countries.

There are serious risks that must be accounted for if large-scale extraction of methane from hydrates is going to happen. It has for instance been suggested that methane hydrate dissociation can destabilize marine sediments, possibly resulting in underwater landslides [67, 77]. Whether such events can be triggered either by global warming from human activities or from trying to extract methane from the hydrate seem to be an open question [53].

2.2 Molecular structure

Different hydrate structures form based on the size of the guest molecules. These structures are characterized by what kinds of cages they contain. The cages are described

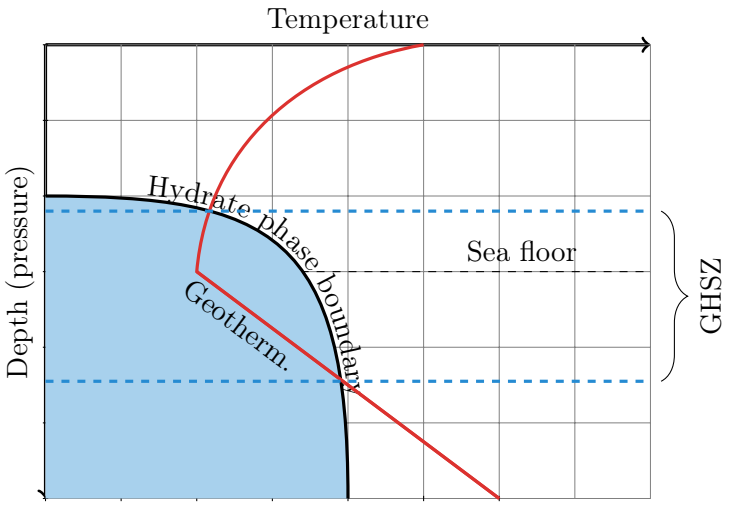


Figure 2.2: Gas hydrate stability conditions in seawater (blue filled area). The red line shows the temperature as a function of depth. In seawater, the temperature falls all the way down to the sea floor, then the temperature follows the geothermal gradient, which means it increases with depth. This leaves a limited depth zone where the temperature–pressure configuration within the stability conditions of gas hydrates. This is the gas hydrate stability zone (GHSZ). Note that the pressure gradient is smaller in seawater than in permafrost, so the GHSZ starts deeper.

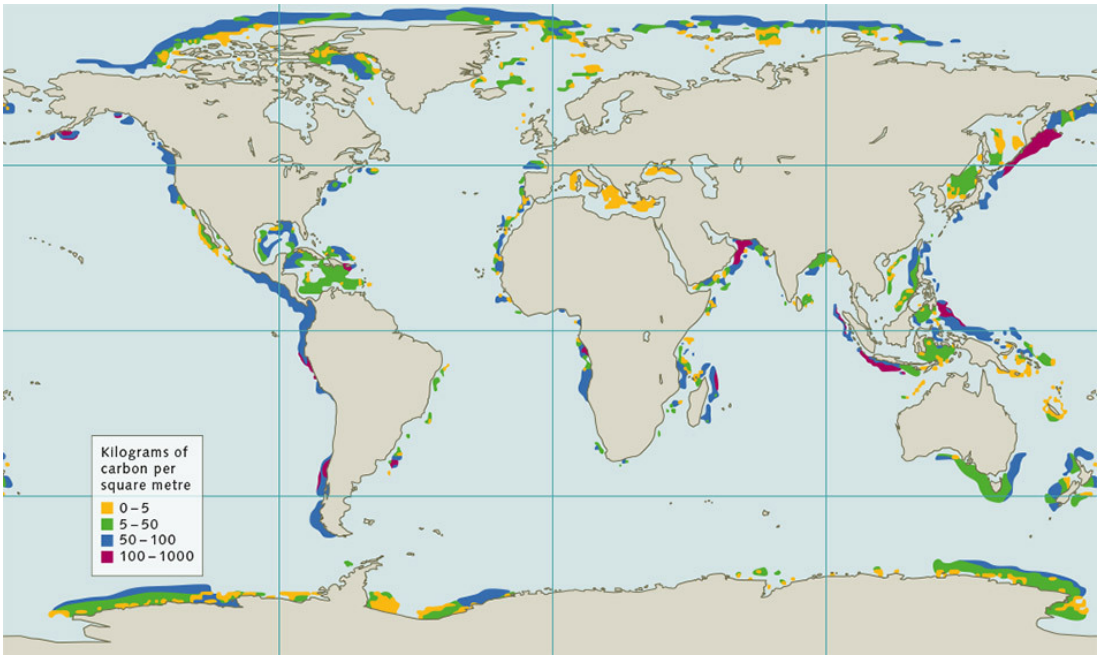


Figure 2.3: Estimated methane hydrate occurrences in the world. This map is taken from the World Ocean Review [13], and shows the data of Wallmann *et al.* [75].

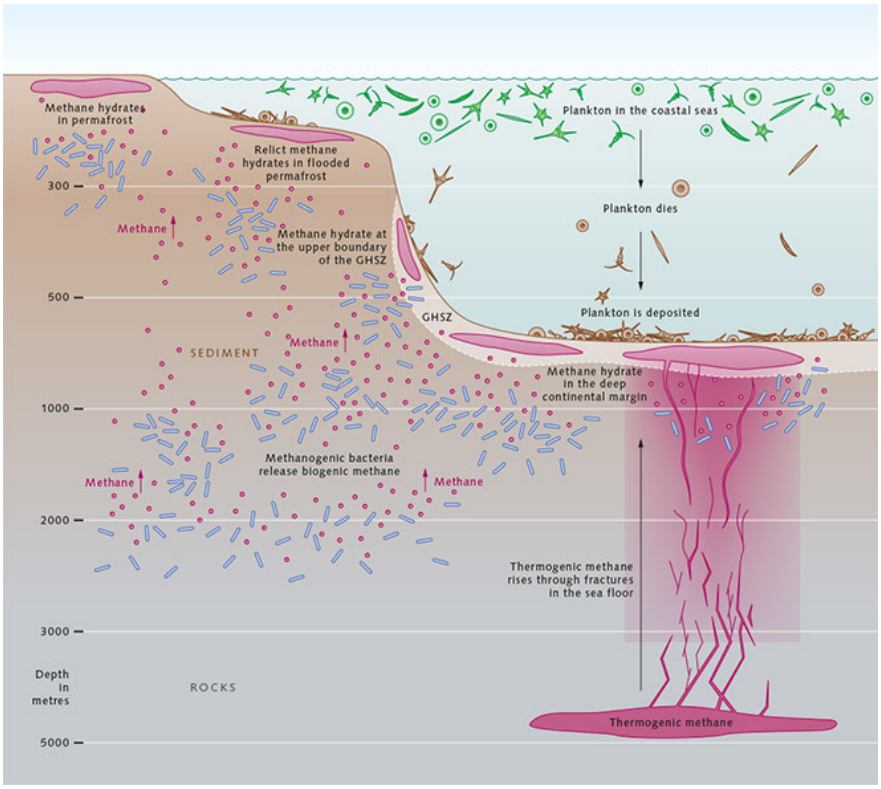


Figure 2.4: Geological setting of methane hydrate formation. This figure is taken from the World Ocean Review [13].

with the notation x^y where x is the number of faces of a certain type, and y describes the type of face by the number of corners on that face of that type. The most common structures for clathrates with one type of guest molecule are the so-called structure I (sI) and structure II (sII). These structures, along with the cages that form them, are illustrated in figure 2.5. The sI structure contains 5^{12} and $5^{12}6^2$ cages in ratio 1 : 3, the sII structure contains 5^{12} and $5^{12}6^4$ cages in ratio 2 : 1.

Small guest molecules, such as methane and ethane, usually form sI hydrates if the conditions for hydrate formation are met [31]. However, that is only certain if the hydrate is either purely methane or purely ethane. For mixtures of methane and ethane forming gas hydrates, either sI or sII can be formed, depending on the relative amounts of methane and ethane [66].

Not all cages need to be occupied by a guest molecule, so a pure methane hydrate sample can, in addition to its cage structure, be characterized by its cage occupancy. It is common to use the hydrate number to describe cage occupancy, which for methane hydrate is

$$CH_4 \cdot n_w H_2O, \tag{2.1}$$

where n_w is the hydrate number. For a fully occupied sI hydrate, the hydrate number would be $n_w = 5.75$. Hydrate numbers have been reported both for laboratory grown methane hydrates and for natural occurring ones. Hydrates grown in laboratory have shown high cage occupancy both with excess water and excess methane, although excess methane yield the highest occupancy. Circone *et al.* [18] report values within 5.9 to 6.1 for a relatively wide range of growth conditions: Pressures from 1.8 to 9.6 MPa and temperatures from 263 to 287 K.

2.3 Mechanical properties from experiments

An understanding of the mechanical properties of methane hydrates is essential for understanding methane hydrate dissociation processes. Under constant temperature and pressure conditions, the methane hydrate has a higher density than the water + free methane system. Therefore, dissociation of some methane hydrate in a reserve will result in a different stress state of the surrounding hydrate. Also, in a conventional production setting, the hydrate will be drilled, which will impose stresses in the hydrate.

It turns out that it is hard to say something general about the mechanical properties. In a review paper from 2012, Ning *et al.* [53] state:

> Few mechanical properties are reported , and their measurements are difficult, partly because it is almost impossible to obtain pure hydrate samples.

This must be taken into account when going through the experimental results on mechanical properties of methane hydrates. It also means that experimental results probably will not be directly comparable to molecular models of pure samples.

Much of the research has been done on hydrate-bearing sediments, and functional relationships have been proposed that relate the strength of a hydrate-bearing sediment to the pure hydrate strength and the hydrate saturation (the amount of hydrate in

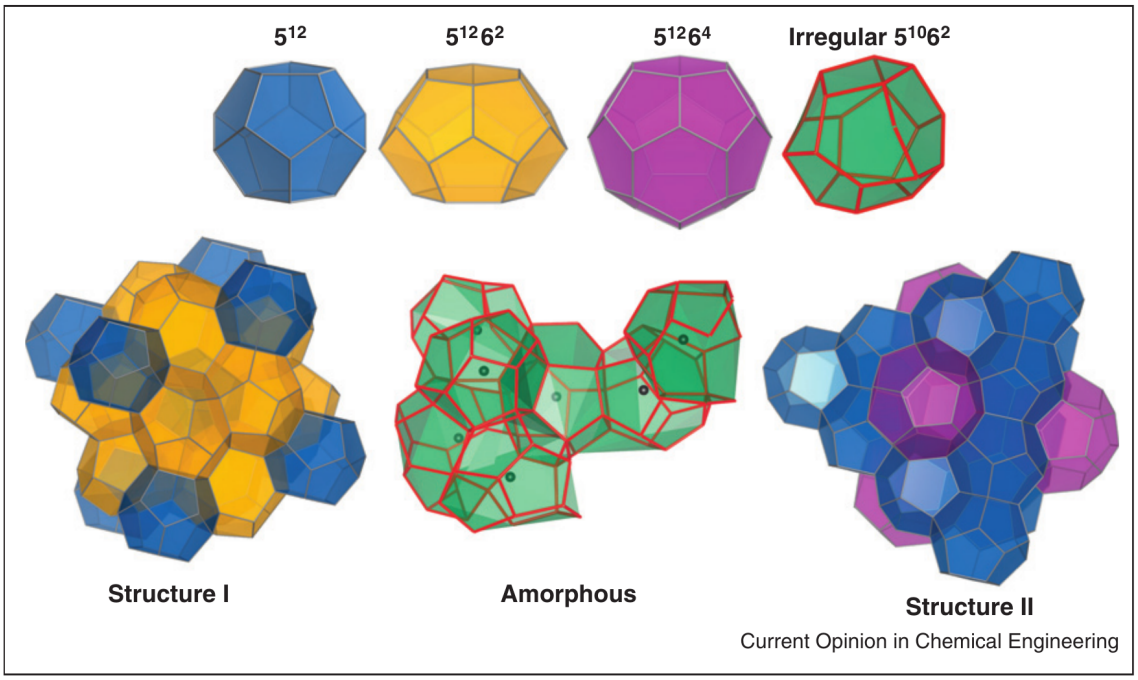


Figure 2.5: Cage structures of single guest methane hydrates occurring in nature. Both structure 1 (sI) and structure 2 (sII) contain the 5^{12} cage. Reprinted from Barnes & Sum [6] with permission from Elsevier.

the sediment). But since the strength of the mechanical properties of the hydrate are uncertain, the assessment of such relationships is hard. Ning *et al.* [53] consider the mechanical properties of pure hydrates essential for understanding the mechanics of hydrate-bearing sediments, which is essential for understanding both how to extract methane and to understand the risks associated with methane extraction from hydrates. Furthermore, almost all strength tests on methane hydrates, tests where the hydrate is subjected to some stress to break it, are axial compression tests, which leaves a limited basis for comparison with simulations.

2.3.1 Typical experimental setup

A main challenge when doing experiments on methane hydrates is that samples of methane hydrate are not stable in room temperature and atmospheric pressure. Therefore, experimental equipment for the study of mechanical properties of hydrates consist of two main parts: A hydrate formation unit, and a measurement unit, so that the hydrate doesn't need to be removed from its stability conditions during measurement. Sometimes, the measurement unit is actually an axial compression chamber, so that axial tests can be done. Figure 2.6 shows an experimental setup for measuring elastic wave speeds [74].

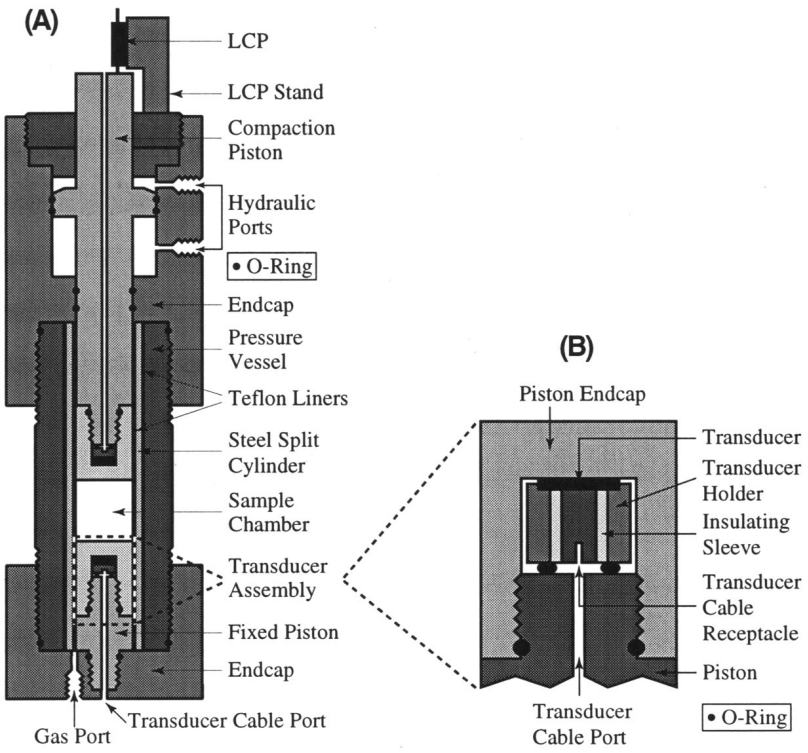


Figure 2.6: Example of an experimental setup. Reprinted from Waite *et al.* [74] with permission from Wiley.

Table 2.1: Mechanical properties of sI methane hydrate as reported in [74].

Property	Value
V_p	$3650(50)\,\mathrm{m\,s^{-1}}$
V_s	$1890(30)\,\mathrm{m\,s^{-1}}$
Poisson's ratio	0.317(6)
Shear Modulus	3.2(1) GPa
Isothermal Young's modulus	7.8(3) GPa

2.3.2 Experimental results

Even though we cannot guarantee the purity of methane hydrate samples used in experiments, there are still robust findings that are valuable pointers for numerical investigations. Waite *et al.* [74] measured mechanical properties using the compressional and shear wave speeds, and assuming the sample to be isotropic and homogeneous. Their results are given in table 2.1.

In the review by Ning *et al.* [53], it is claimed, based on numerous experiments they reviewed, that the tendency for the compressive strength of methane hydrates, is that it increases with increasing confining pressure and decreasing temperature. But it is also stated that all the other mechanical properties depend highly on the temperature, the pressure, the cage occupancy etc., which means results from experiments under slightly different conditions are hard to compare. A surprising observation with regards to compressional strength, is that methane hydrates exhibit strain-hardening for compressional strains as high as around 15-20 % [22, 64], which is very high compared to regular water ice. A strain-hardening curve is shown in figure 2.7.

2.4 Classical potential models of water

Water is a fundamental aspect of life on earth, but it is surprisingly difficult to model all aspects of water within the same modeling framework. Water is difficult to model, and many models are developed in the pursuit of understanding water in all its phases and for various types of systems.

The water molecule can be represented in many different ways, and the most common representation is the ball–stick model of two hydrogen atoms connected to an oxygen atom. In this representation, the parameters for the distance between the oxygen and the hydrogen and the angle between them can been found either experimentally or from quantum mechanics methods such the as the Hartree-Fock method [25, 63]. Data the the geometry of water is given in table 2.2. Another representation of the water molecule is by its wave-function, which can be visualized for example by the electron density around the nuclei of the molecule. Such a representation can be derived from quantum mechanics. Both representations are illustrated in figure 2.8. The quantum mechanic description is too complex to be directly applied in molecular dynamics, so the molecular dynamics representation must be some variation over the connected-particles picture.

Probably the first to model water using the ball-stick approach was the model of

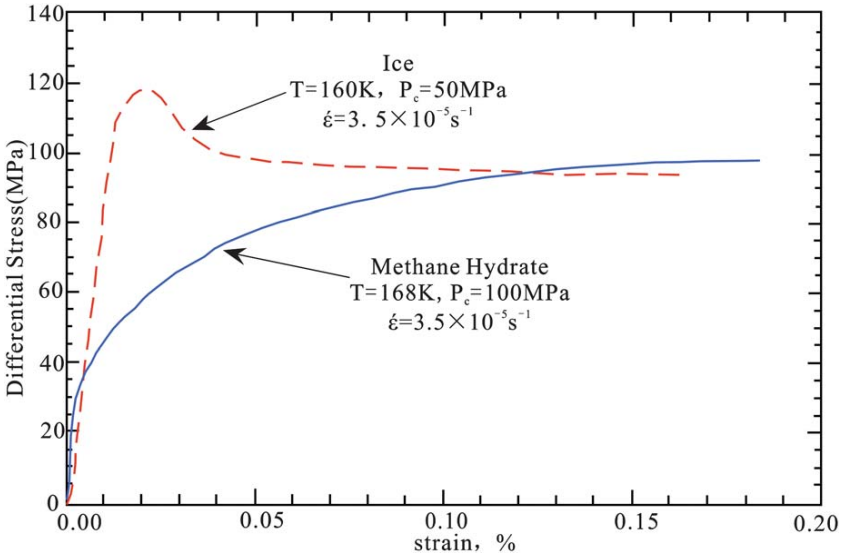
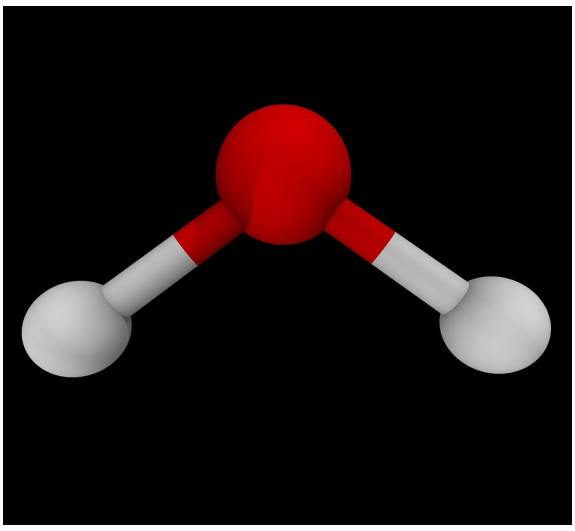
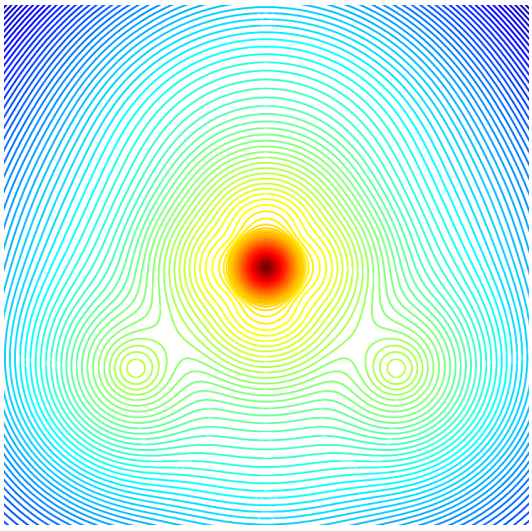


Figure 2.7: Differential stress–strain curve for methane hydrate and water ice. Note that the axis text on the first axis is wrong. The numbers are not supposed to be in percent, which means the axis range is 0 to 20 %. Reproduced from [53] with permission of The Royal Society of Chemistry.

Bernal & Fowler [10] form 1933. The Bernal & Fowler model is a four-site model where the oxygen and hydrogen nuclei take their positions form the best real-water estimates, but the oxygen-charge is moved slightly along the bisector of the HOH-angle. This model inspired (and is essentially equal to) the TIP4P model of Jorgensen *et al.* [41], which is one of the most widely studied water models today. Other popular models are the TIP3P model [41], a three-particle model where the oxygen charge is placed on the oxygen nucleus, and the SPC water model [9], another three-particle model, where the HOH-angle is optimized for tetrahedral configurations rather that being the correct water molecule configuration. These models come in both flexible and rigid versions, and the rigid versions are often preferred since they reduce the computational cost of simulations. It has also been commented that the hydrogen vibrations are too fast to be treated with classical mechanics [72], rendering flexible approaches to the water molecule problematic in non-quantum mechanical models. Both the SPC, TIP3P and TIP4P models are empirically fitted models, which means the models will reproduce some features of water by design. The parameters have gotten better over the years, and Vega & Abascal [72] note that the best current parameter set for the TIP4P-model, TIP4P/2005, is probably close to the best possible overall performance of a 4-particle classical potential model of water. To improve the performance, quantum effects must be accounted for in some way.



(a) The water molecule represented with its individual atoms connected with bonds.



(b) Electron density representation of the water molecule. The density is calculated by the Hartree-Fock method. Taken from my simulations for a course at the University of Oslo.

Figure 2.8: Two representations of the water molecule.

Table 2.2: Geometry of the water molecule [24].

Description	Symbol	Value
H-O-H angle	θ	104.52°
Distance O-H	d_{OH}	0.9572 Å

2.5 Molecular dynamics modeling of methane hydrates

Due to the lack of good experimental results, numerical modeling can be important to investigate methane hydrates. One of the ways to model them is through molecular dynamics simulation, where trajectories of individual atoms are calculated using classical equations of motion.

There are essentially two choices to be made when designing a molecular dynamics simulation: What potential models to use, and what system to simulate. By system, I mean both the initial condition and the conditions during simulation (temperature, pressure, etc.).

For the potential models of methane hydrates, there are two common strategies: All-atom potentials and united-atom potentials. In the all-atom potentials, methane and water are represented by all of its atoms. Interactions between atoms belonging to the same molecule are bonding, and interactions between atoms belonging to different atoms are non-bonding. In the united-atom potentials, all atoms in a molecule are represented by one particle. Interactions between molecules in united atom representations are non-bonding, and the functional form of the potential is usually more complicated

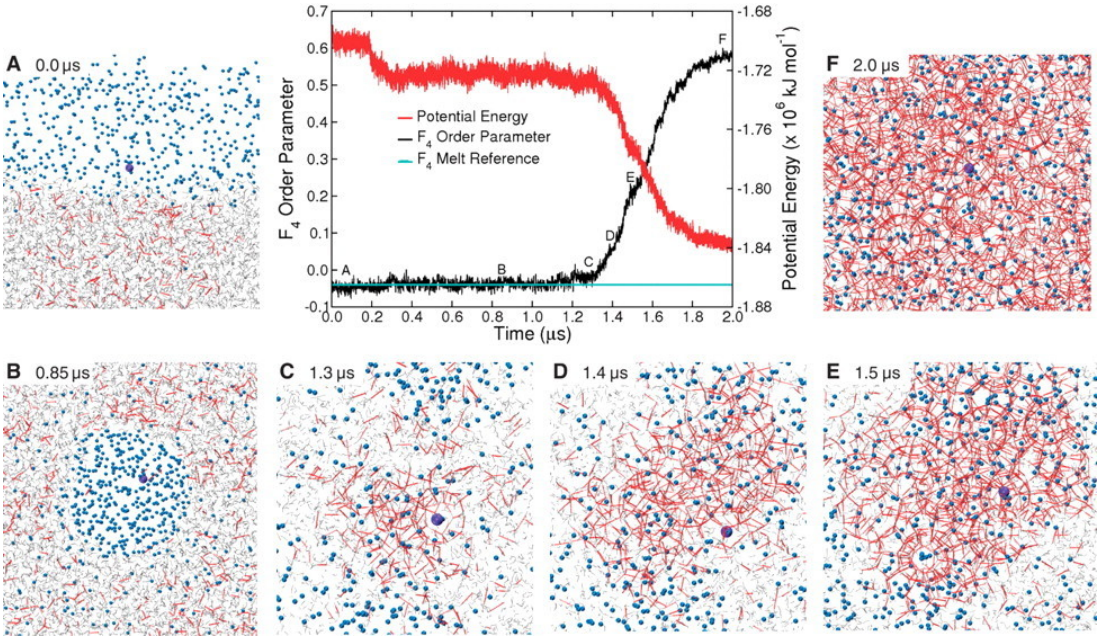


Figure 2.9: Nucleation and growth of methane hydrates. From Walsh *et al.* [76]. Reprinted with permission from AAAS.

than for all-atom models.

In the first successful simulation of methane hydrate nucleation, Walsh *et al.* [76] used a combination of these strategies. The water model was an all-atom model, TIP4P/Ice (described in detail in chapter 5), and the methane model was a united-atom model, united-atom methane, which is a pure Lennard-Jones potential. Figure 2.9 shows illustrations of the formation process. Later, Jacobson & Molinero [39] proposed a coarse-grained model using the united-atom approach both on the water model and the methane model. This model has been used in several studies looking at nucleation and growth of methane hydrates.

Potentials will be further discussed in chapter 5.

2.6 Quantum mechanical calculations on methane hydrates

The properties of the sI hydrate cage have been calculated using DFT-analysis. The results differ quite a lot from the experimental values, but since the experimental values are uncertain, it is hard to assess the value of the DFT-analysis.

There have also been some efforts on fitting a methane-water potentials using ab initio quantum mechanical methods. A popular method is by second order Møller–Plesset perturbation theory (MP2). Anderson *et al.* [4] used this to develop a potential for argon–water and methane–water interactions and applied the methane–water results on methane hydrates. They fix the internal configuration of each molecule, i.e. the hydrogen positions relative to the oxygen in water and the hydrogen positions relative to carbon in methane. Then they map the six-dimensional energy surface of the methane–

water interaction. The dimensions are the C–O distance and five angles describing the relative rotations of the molecules. One of the conclusions is that the Exponential–6-potential better represents the $H_4\mathbf{C}$–$\mathbf{O}H_2$ interaction than the Lennard–Jones potential. That study does not cover the interactions between particles of the same species (e.g. water–water).

2.7 Molecular dynamics modeling of fracture

Significant work has been invested to characterize the fracture of relatively simple crystal lattices such as the FCC-lattice. Abraham *et al.* [2] studied fracture in two-dimensional notched solids using the Lennard-Jones potential to model brittle material and the EAM potential [20] to model ductile material. They do this to investigate general features of a large class of fracture problems, rather than studying properties of a specific substance. Specifically, they observe that the crack tip in brittle fracture becomes unstable when the crack tip speed (v_c) reaches around a third of the Rayleigh wave speed (v_R). This is somewhat similar to observations of Zhou *et al.* [79] from two-dimensional simulations; that the crack must reach a speed of more than one third of the Rayleigh speed to be able to branch. Later, they also studied ductile fracture, and observed characteristic *dislocation loops* near the crack tip. Holland & Marder [33] studied brittle failure of silicon using the potential of Stillinger & Weber [65]. Buehler & Gao [15, ch. 6] reviews dynamical fracture of homogeneous lattices, and adds to the observation by Abraham (instability for $v_c > 1/3v_R$) that the instability velocity depends not only on the Rayleigh wave speed, but also on hyper-elastic (non-linear elastic) properties of the material. Regarding the crack surface; stable, low speed cracks produce mirror interfaces, whereas unstable cracks form either mist (flat but rough) or hackle regions (rough) on the crack surface. In the studies cited in Buehler & Gao [15, ch. 6], tensile strain is applied on the boundary by forcing the position of the atoms in the edge planes parallel to the crack plane.

Hantal *et al.* [30] studied the fracture toughness of illite using clayFF and reaxFF. The actual fracture propagation was not studied – only the fracture toughness and the stress–strain curves. In that study, the tensile strain was applied in discrete steps by expanding the simulation box normal to the crack plane and remapping all particle positions.

I have not been able to find any studies on molecular dynamics simulations of fracture in methane hydrates.

2.8 Research questions

Since I have not been able to find any studies on fracture of methane hydrates using molecular dynamics, the expectations of what can be done during a master project are limited. Below, I name a few questions concerning methane hydrates that seem within reach to address:

1. What is the fracture toughness of methane hydrates in simulations? This question is not only interesting to compare with experiments, but to complement experi-

ments. Fracture toughness of materials are usually estimated with standardized mechanical tests, but since such tests are hard to perform on methane hydrates, it is possible that simulation results can complement experimental results, and be important for instance as the pure-hydrate strength going into calculations of the strength of hydrate-bearing sediments.

2. Is pure methane hydrate ductile or brittle, and is the brittle- or ductileness dependent on the strain rate the sample is subjected to?

3. What does the fracture surface look like? Is it mirror-like or hackle-like? Does the fracture surface develop in time after fracture propagation?

4. How much methane hydrate is dissociated during fracture? How much methane is freed?

5. How predictable are the fracture properties of methane hydrates predictable? Is it such that for a given stress or strain applied to a piece of methane hydrate, the critical stress/strain or the time it takes before it starts cracking can be accurately estimated? Or is it a statistical processes with a wide waiting-time distribution that govern the fracture initiation – or even propagation?

6. Compressional strain hardening up to almost 20 % strain. How can this be explained?

Based on these questions, I will produce novel results and insights on molecular dynamics modeling of methane hydrate fractures. Question 1 and 2 will be answered quite conclusively (but only within the model I choose). Question 3 will be discussed along with question 4, offering some descriptive results but little insight. Question 5 results in a claim that I hope to be able to verify in the future. Question 5 will be ignored.

There are also questions regarding how to model methane hydrates, and event within molecular dynamics modeling of methane hydrates, little is actually known – especially when it comes to fracture.

1. What potential best reproduce fracture of methane hydrates?

2. How may cracks be triggered in simulation if the results are going to say something about reality?

These two questions are listed mostly for completeness, as this work is not about potentials itself. However, understanding the limitations of the potentials is essential when doing molecular dynamics, and I also have to choose a potential. Question 2 will be touched, especially with regards to the limitations of molecular dynamics simulations.

Chapter 3

Elasticity and failure

To have a framework to discuss failure and fracture in methane hydrates, I will introduce some theory of elasticity and failure in linear elastic materials. This will also be needed in order to be explicit about how stresses and strains are imposed on the model systems.

3.1 Linear elasticity

In general, methane hydrates are anisotropic materials. However, they are sufficiently isotropic to be treated as isotropic in this work. I will start by introducing the general tensor form of Hooke's law, and then provide the simplifications resulting from looking at an isotropic material. This presentation will take Hooke's law as a given, but it can be derived for example with an energy approach as in Buehler [14, p.105].

3.1.1 Stress and strain

Linear elasticity is based on the idea that deformations of a material will result in a linear reaction in terms of forces from that material. *Strain*, ϵ, is a deformation, and the *normal* strain is defined as the relative elongation of a body;

$$\epsilon = \frac{l - l_0}{l_0} \tag{3.1}$$

Where l_0 is the equilibrium length of that body and l is the final length. Additionally, the body can be subjected to *shear strain*. Shear strain can be defined as the in-plane component of the difference in midpoint position between two facing end planes of an infinitesimal cubic element divided by the equilibrium distance between these planes.

It is common to use an index notation that clarifies on what plane a strain is present, and in which direction. Strains are denoted ϵ_{ij} where the first index says what planes are involved in the strain (the index denotes a vector perpendicular to the plane), and the second index says in what direction the strain is present. For instance: ϵ_{yy} is the relative elongation of the distance between the xz end-planes of an infinitesimal cubic element, whereas ϵ_{xy} is the y-component of the distance between the midpoint of the left and right yz-planes divided by the x-component of the equilibrium distance between these points.

Stress is a force density acting on a body. Uniform normal stress on a plane is defined as:

$$\sigma_n = \frac{F_\perp}{A} \tag{3.2}$$

Shear stress is the force parallel to the plane:

$$\sigma_s = \frac{F_\parallel}{A} \tag{3.3}$$

Like for strains, stress on different planes and directions of an infinitesimal cube are indexed like σ_{ij} where i says what plane the force acts in, and j in what direction the force acts.

3.1.2 Hooke's law

Stresses cause strains and vice versa. This is reflected in the generalized Hooke's law:

$$\sigma_{ij} = c_{ijkl}\epsilon_{kl} \tag{3.4}$$

This law relates the Cauchy stress tensor σ_{ij} to the strain tensor ϵ_{kl} in a linearly elastic material. All material properties are contained in the stiffness tensor c_{ijkl}. In this form, Hooke's law basically states that each component of the Cauchy stress tensor depends linearly on *all* components of the strain tensor.

Hooke's law contains two tensors of rank 2 with 9 components each, and one tensor of rank 4 with 81 components. Fortunately, there are symmetries to be exploited. First, the Cauchy stress tensor is symmetric, which leads to $c_{ijkl} = c_{jikl}$. Second, the strain tensor is symmetric, so $c_{ijkl} = c_{ijlk}$. This means we are left with 6 independent combinations of each of ij and kl, and a total of 36 components.

Using the rank reduction method of Voigt ([73], or any standard book on elasticity), the stress and strain matrices can be written as vectors:

$$\underline{\sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{pmatrix} \rightarrow \boldsymbol{\sigma} = \begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{pmatrix} \equiv \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} \tag{3.5}$$

$$\underline{\epsilon} = \begin{pmatrix} \epsilon_{11} & \epsilon_{12} & \epsilon_{13} \\ \epsilon_{21} & \epsilon_{22} & \epsilon_{23} \\ \epsilon_{31} & \epsilon_{32} & \epsilon_{33} \end{pmatrix} \rightarrow \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\epsilon_{23} \\ 2\epsilon_{13} \\ 2\epsilon_{12} \end{pmatrix} \equiv \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{pmatrix} \tag{3.6}$$

For the same reasons, the stiffness tensor can be reduced to rank 2 (I choose not to

write out the complete stiffness tensor, only the reduced one):

$$\mathbf{C} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{31} & C_{31} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} \end{pmatrix} \tag{3.7}$$

Hooke's law can now be written as a matrix equation:

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\epsilon} \tag{3.8}$$

It actually turns out that the stiffness matrix is symmetric, because stress and strain are work-conjugates:

$$\sigma_i = \frac{\partial u}{\partial \epsilon_i} \tag{3.9}$$

Where u is the energy volume density associated with the stress-strain configuration. Inserting this into Hooke's law gives:

$$C_{ij} = \frac{\partial^2 u}{\partial \epsilon_i \partial \epsilon_j} = \frac{1}{V}\frac{\partial^2 U}{\partial \epsilon_i \partial \epsilon_j} \tag{3.10}$$

This relation can be used to calculate the stiffness matrix from any simulation of an elastic material where strains can be imposed and potential energy can be measured.

When representing the stress and strain matrices in practice, it is common to distinguish between stress and strain contributions. In standard Cartesian coordinates, equations 3.5 and 3.6 turn into:

$$\underline{\sigma} = \begin{pmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{pmatrix} \rightarrow \boldsymbol{\sigma} = \begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \tau_{yz} \\ \tau_{xz} \\ \tau_{xy} \end{pmatrix} \tag{3.11}$$

$$\underline{\epsilon} = \begin{pmatrix} \epsilon_{xx} & \epsilon_{xy} & \epsilon_{xz} \\ \epsilon_{yx} & \epsilon_{yy} & \epsilon_{yz} \\ \epsilon_{zx} & \epsilon_{zy} & \epsilon_{zz} \end{pmatrix} \rightarrow \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \epsilon_{yz} \\ \epsilon_{xz} \\ \epsilon_{xy} \end{pmatrix} \tag{3.12}$$

Where the shear stress components have changed from σ to τ. This is the notation I will use.

3.1.3 Isotropic materials

An isotropic material is a material where the stiffness properties do not depend on space directions in the material. Isotropic materials can be described by two independent parameters, for example Young's modulus E and Poisson's ratio ν. Young's modulus can be defined as:

$$E = \frac{\sigma_{xx}}{\epsilon_{xx}} \tag{3.13}$$

Where σ_{xx} is a normal stress applied in the same direction as the normal strain ϵ_{xx} is measured. This strain is known as *normal strain*. Since the directions don't matter in isotropic materials, I have chosen to define Young's modulus along the x-axis. When an isotropic material is subjected to tensile stress and becomes longer, it will simultaneously contract in the directions normal to the applied tension. This is known as *lateral strain*. The ratio of lateral strain to normal strain is known as *Poisson's ratio*:

$$\nu = -\frac{\epsilon_{yy}}{\epsilon_{xx}} = -\frac{\epsilon_{zz}}{\epsilon_{xx}} \tag{3.14}$$

As promised, these two properties shall completely describe the elastic material. The stiffness tensor of an isotropic material is:

$$\mathbf{C} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & (1-2\nu)/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & (1-2\nu)/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (1-2\nu)/2 \end{pmatrix} \tag{3.15}$$

3.1.4 Plane strain and plane stress

There are conditions where a three-dimensional elastic problem can be analyzed as a two-dimensional problem. This is useful for example when applying standard results of linear elastic fracture mechanics, since many of those results are for two-dimensional systems. Plane strain and plane stress are such conditions. The formal definitions of plane strain and plane stress are given in equations 3.16 and 3.17, respectively.

$$\underline{\underline{\epsilon}} = \begin{pmatrix} \epsilon_{11} & \epsilon_{12} & 0 \\ \epsilon_{21} & \epsilon_{22} & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{3.16}$$

$$\underline{\underline{\sigma}} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & 0 \\ \sigma_{21} & \sigma_{22} & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{3.17}$$

For plane strain, we allow a nonzero stress component σ_{33}. Likewise, for plane stress we allow a nonzero strain component ϵ_{33}. However, these components can only be results of the analysis, they shall not enter the analysis.

3.1.5 Elastic moduli

As mentioned earlier, an isotropic material can be uniquely described by two constants. They can for example be two of the following (there are other possible constants):

- Bulk modulus, K
- Young's modulus, E
- Lamé's first parameter, λ
- Shear modulus, G
- Poisson's ratio, ν
- P-wave modulus, M

Below, I write the definitions of bulk modulus and shear modulus and give their relation to Young's modulus and Poisson's ratio, as this is useful to calculate elastic wave velocities in materials.

The shear modulus is defined as the ratio of shear stress to shear strain:

$$G = \frac{\sigma_{ij}}{\epsilon_{ij}}, \qquad i \neq j \tag{3.18}$$

The shear modulus can be related to Young's modulus and Poisson's ratio by:

$$G = \frac{E}{2(1+\nu)} \tag{3.19}$$

The bulk modulus K is defined as:

$$K = \rho \frac{\mathrm{d}P}{\mathrm{d}\rho} \tag{3.20}$$

Where P is hydrostatic pressure, $P = \frac{1}{3}(\sigma_{xx} + \sigma_{yy} + \sigma_{zz})$, when $\sigma_{xx} = \sigma_{yy} = \sigma_{zz}$. ρ is the mass density of the material. The bulk modulus can be related to Young's modulus and Poisson's ratio by:

$$K = \frac{E}{3(1-2\nu)} \tag{3.21}$$

3.1.6 Elastic waves

In homogeneous isotropic materials, one can observe shear waves and pressure waves. Shear waves depend on the shear modulus and travel with a speed of:

$$v_s = \sqrt{\frac{G}{\rho}} = \sqrt{\frac{E}{2\rho(1+\nu)}} \tag{3.22}$$

Where ρ is the mass density of the material.

Pressure waves depend on the bulk modulus and travel at a speed of:

$$v_p = \sqrt{\frac{K}{\rho}} = \sqrt{\frac{E}{3\rho(1-2\nu)}} \tag{3.23}$$

These relations can be used to find elastic properties from acoustic measurements. That is convenient if a material of unknown elastic moduli is hard to put in an apparatus for deformation. These relations are also convenient for finding elastic wave velocities from molecular dynamics simulations, as the moduli are easy to measure.

3.2 Linear elastic fracture mechanics

Linear elastic fracture mechanics (LEFM) deals with the fracture of linear elastic materials. The simplest version of this theory assumes that fracture is governed by two equivalent properties: The stress intensity factor K, and the energy release rate G. Here, I introduce the background for these concepts.

3.2.1 Brittle and ductile materials

In materials science, it is common to distinguish between brittle and ductile materials. A material i brittle if it "breaks without significant strain" (Wikipedia, brittle). Brittleness can also be described as "an inability to deform plastically" [7], whereas Ductility is a solid material's ability to deform plastically [7]. It is actually hard to come along a more satisfying definition – ductility and brittleness seem to be the kind of knowledge that everyone in the field have, but no one writes down as an equation. Throughout this thesis, I will use the term brittle – by which I will mean ideally brittle – in two ways. First, the intuitive definition: A material is brittle if it can be elastically deformed and suddenly breaks over essentially no change in the strain applied to it. Secondly, a definition that will be clearer later in this section: A material is brittle if the strain energy needed to break it is only slightly higher than the energy needed to open the projected crack surface of the crack that is created when the material fails. An illustration of brittle and ductile fracture in terms of potential energy stored in the system during loading and failure is shown in figure 3.1.

3.2.2 Modes of loading

There are three different modes of crack separation: I ='opening', II = 'sliding', III = 'tearing'. These are illustrated in figure 3.2. The presentation from here on will only consider mode I loading – the opening mode.

3.2.3 Failure criteria

As is many fields, the first recorded studies are the ones of Leonardo da Vinci. da Vinci discovered that short iron wires are stronger than long iron wires. A common interpretation is that random flaws make the iron wires weaker at some points, and that a longer wire has a higher probability of a weak spot than a long one. More precisely:

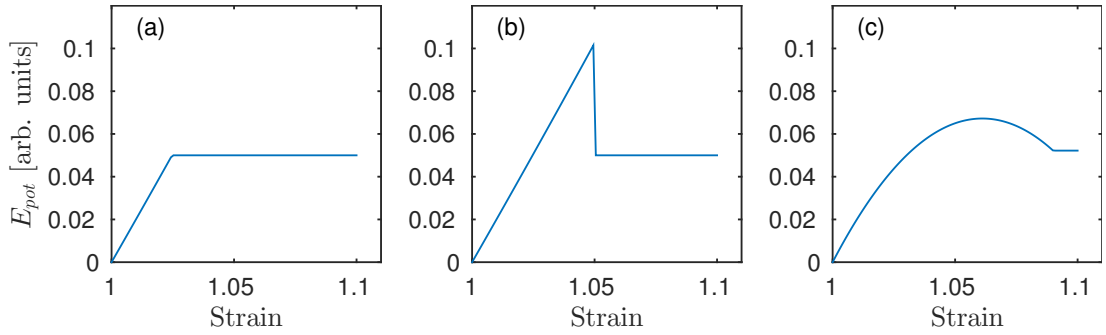


Figure 3.1: Potential energy as a function of applied strain for systems held at a constant temperature. The figure shows three idealized examples of failure. (a) and (b) are brittle, (c) is ductile. In (a), the system is loaded adiabatically and reaches a stress state where $\mathcal{G}_c = 2\gamma_s$ and breaks. No energy is lost to plastic deformation or heat. In (b), the system is loaded isothermally and breaks at $\mathcal{G}_c > 2\gamma_s$. There is no plastic deformation, but heat flows in and out of the material. In (c), the system is continuously deforming plastically through the straining process – the material is very ductile. Note that it is not possible to see the amount of energy lost to plastic deformation from (c). The plateau in all figures represents a state where a crack propagated through the whole system – the system is divided into two parts – so additional straining does not contribute potential energy.

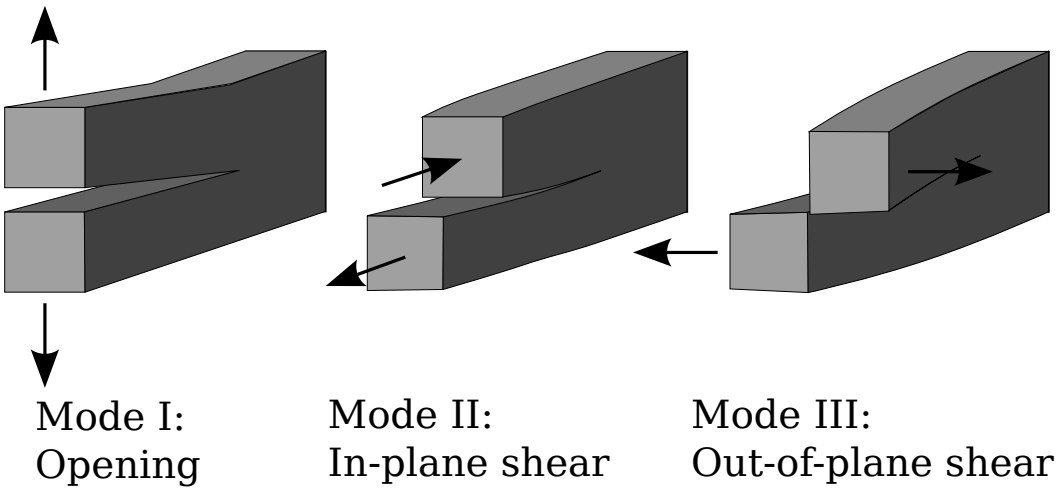


Figure 3.2: Three modes of crack separation. ("Fracture modes v2" by Twisp. Licensed under Public Domain via Wikimedia Commons)

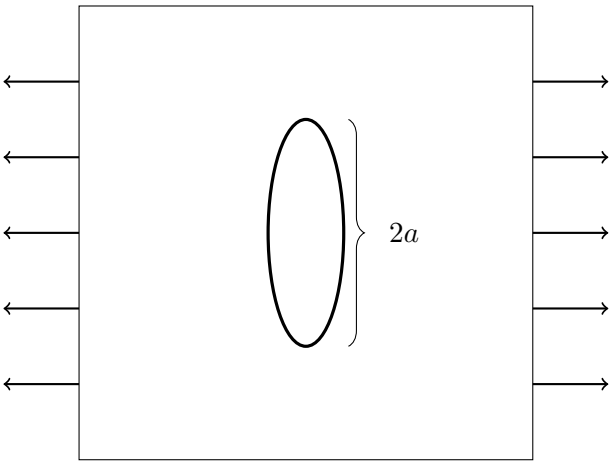


Figure 3.3: Elliptical crack in an "infinite" plate.

The strength will be governed by the largest flaw, and the expected size of the largest flaw grows with the wire length. A major goal of LEFM is to be able to predict the failure of structures: what is the pressure required to break a sample of a material? The first somewhat successful attempt to predict failure was the efforts of Inglis [37]. He introduced the concept of stress concentration at a crack tip to find the fracture toughness of brittle materials. Specifically, he found that for an elliptical crack in an infinite sheet (figure 3.3, the local stress at the crack tip as a function of the faraway tensile stress was:

$$\sigma_{\text{cracktip}} = 2\sigma_{\text{faraway}}\sqrt{\frac{a}{\rho}} \tag{3.24}$$

Where a is half the length of the major axis of the ellipse, and ρ is the radius of curvature of the ellipse close to the crack tip. The theory of Inglis is an atomistic one, and he required the stress concentration at the crack tip for failure to be the stress needed to break atomic bonds. The theory is described in Anderson [5, p.27]. The expression for this can be found by regarding the pressure exerted between two atoms when they are dragged away from their equilibrium configuration as a sine function with a maximum stress equal to the cohesive strength σ_c of the bond:

$$\sigma = \sigma_c \sin\frac{\pi x}{x_0} \tag{3.25}$$

Where x_0 is the typical distance between atoms. If the origin is set to the equilibrium distance x_0, and the force constrained to act for another equilibrium distance, then the energy area density to break bonds is:

$$u_{bond}^{area} = \int_0^{x_0} \sigma_c \sin\frac{\pi x}{x_0}\mathrm{d}x = 2\sigma_c\frac{x_0}{\pi} \tag{3.26}$$

It is now convenient to introduce a new quantity, the surface energy γ_s. This is the mechanical energy needed to create new crack surface area. This is exactly half the energy area density needed to break an atomic bond, since breaking atomic bonds create

two surfaces. The surface energy for this simple bond stress model is then:

$$\gamma_s = \sigma_c \frac{x_0}{\pi} \tag{3.27}$$

Young's modulus should (from its definition) be the slope of the stress with respect to strain. The slope of the stress with respect to distance in equation 3.25 is $\frac{\pi\sigma_c}{x_0}$, which means Young's modulus is:

$$E = \pi\sigma_c \tag{3.28}$$

Combining equations 3.27 and 3.28 one obtains an estimate of the cohesive strength of atom bonds using only *one* atomic scale parameter, x_0:

$$\sigma_c = \sqrt{\frac{E\gamma_s}{x_0}} \tag{3.29}$$

Setting the crack tip stress from equation 3.24 equal to the critical local stress from equation 3.29, and assuming that the radius of curvature is equal to the distance between atoms (this turns out to be a good assumption, and removes the atomic scale parameter), we get an expression for the critical faraway stress level:

$$\sigma_f = \sqrt{\frac{E\gamma_s}{4a}} \tag{3.30}$$

This is the Inglis formula for the fracture stress on a large sheet with a crack of width $2a$.

In 1920, Griffith improved on the flaw-approach [28], but instead of building a theory from the atomic level, he assumed the following: A crack will propagate from a flaw if the strain energy that will be released during crack growth is higher than the corresponding surface energy associated with the created crack surface. The surface energy density is denoted γ_s and have units of energy per area. Griffith's approach works well for ideally brittle materials. The formula for critical faraway stress with Griffiths theory is:

$$\sigma_f = \sqrt{\frac{2E\gamma_s}{\pi a}} \tag{3.31}$$

For an elliptic crack of width $2a$ (Same conditions as with Inglis' theory). This expression is actually equal to the expression of Inglis, except for a factor, even though Griffith's theory is based solely on continuum mechanics.

For a sheet of finite width, the fracture stress is slightly lower since the crack weakens the material (the cross-sectional area bearing the stress is shrinking). The exact expression for a sheet of width $2W$ is:

$$\sigma_f = \sqrt{\frac{2E\gamma_s}{2W\tan\left(\frac{\pi a}{2W}\right)}} \tag{3.32}$$

For reference, I also include the formula for the fracture stress of a penny-shaped crack subjected to remote tensile stress:

$$\sigma_f = \sqrt{\frac{\pi E\gamma_s}{2(1-\nu^2)a}} \tag{3.33}$$

Irwin refined Griffith's approach, and introduced the *energy release rate* rate [38], $\mathcal{G}$. The energy release rate is a property of the elastic state of a linearly elastic material. According to Irwin, a crack will propagate when $\mathcal{G}$ becomes larger than a material-specific value $\mathcal{G}_c$ – the *critical* energy release rate. This differs from Griffith's theory since the critical energy release rate doesn't necessarily have to be the same as the surface energy. For a straight crack of width $2a$ (no longer required to be elliptic) on an infinite plate subjected to tensile stress, the energy release rate is:

$$\mathcal{G} = \frac{\pi\sigma^2 a}{E} \tag{3.34}$$

This is purely a relation concerning how a linear elastic material distributes energy during crack opening. In the particular case of a straight crack on an infinite sheet, we see that the longer the crack, the higher the energy release rate. This fits with the intuition that a larger flaw will reduce the strength of a material. Note that the material doesn't get weaker because the flaw reduces the load-bearing area (the sheet is infinite). The material gets weaker because a longer crack increases the elastic energy that gets released per crack area grown. The critical value can be measured experimentally using a sample with an artificial flaw whose length is known, and measure the yield pressure.

Notice that the formula for energy release rate is equivalent to the Griffith formula, equation 3.32, except that the surface energy $2\gamma_s$ is swapped with the energy release rate.

Irwin also introduced another property, which is equivalent to the energy release rate, namely the *stress intensity factor*, K. The stress intensity factor is a constant of proportionality between the applied stress on a crack and the stress distribution around the crack tip. This is very similar to the approach of Inglis, but it not only concerns the strength of a single atomic bond, but rater the whole stress distribution. So, unlike Inglis' theory, the stress intensity factor is a continuum property. For mode I loading, the stress intensity factor K_I is defined by:

$$\lim_{r\to 0} \sigma_{ij}^I = \frac{K_I}{\sqrt{2\pi r}} f_{ij}(\theta) \tag{3.35}$$

Where r is the distance from the crack tip and θ is the angle from the crack axis. Both the energy release rate and the stress intensity factor concern the distribution of a remote stress, but the stress intensity factor says how the stress itself is distributed near the crack tip, whereas the energy release rate says how much mechanical energy will be released if new crack surface opens.

Like the energy release rate, the stress intensity factor can take a critical value; the *critical stress intensity factor* K_c. Under mode I loading it is called K_{Ic}. This is the standard measure of fracture toughness.

For the sake of history, a quick and dirty summary of the development of linear elastic fracture mechanics: da Vinci's random flaws inspired Inglis to calculate the stress concentration around sharp cracks in an atomistic approach. Griffith thought an energy balance was a better, and that earned him a pre-factor. He created the same theory as Inglis, but got rid of the atomistic view. Irwin decoupled the theory from the actual surface energy and also introduced the stress intensity factor, which is almost like the

Inglis' stress concentration, but it is more rigorously defined and fits in a continuum theory.

3.2.4 Stress intensity factors in anisotropic materials

This section was originally written in case the methane hydrate was to be analyzed as an anisotropic material. That will not happen, but the section is kept to show how quickly the calculations get messy when considering anisotropic materials. Having obtained the fracture area and the energy release rate. A generalized Irwin formula can be used to calculate the stress intensity factor [43]:

$$\mathcal{G} = \pi \mathbf{K}^T [\mathbf{H}] \mathbf{K} \tag{3.36}$$

Where $\mathbf{H}$ is a matrix that depends on the elastic properties of the material. This expression is valid for a plane crack propagating in two opposite directions with symmetric load with respect to the two directions of crack propagation. In the case of mode I loading, only one of the elements of the $\mathbf{H}$-matrix needs to be known, H_{11}. This matrix element was worked out by Laubie & Ulm [43], and is:

$$H_{11} = \frac{1}{2\pi} \sqrt{\frac{C_{11}}{C_{11}C_{33} - C_{13}^2} \left(\frac{1}{C_{44}} + \frac{2}{C_{13} + \sqrt{C_{11}C_{33}}} \right)} \tag{3.37}$$

For isotropic materials under mode I loading, we recover a more familiar expression; Irwin's formula for plane strains:

$$K_{Ic} = \sqrt{\frac{E\mathcal{G}_c}{1 - \nu^2}} \tag{3.38}$$

3.3 Stress concentrations around an elliptical hole

There are several analytical solutions for stress concentrations for isotropic materials with failures of specific geometries. A particularly interesting solution for my purposes is the stress concentration near the crack tip of an elliptic crack in an infinitely large sheet of linearly elastic and isotropic material [5]. I give the equations for a crack along the y-axis with a stress applied along the x-axis.

$$\sigma_{xx} = \frac{K_I}{\sqrt{2\pi r}} \cos\left(\frac{\theta}{2}\right) \left[1 + \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{3\theta}{2}\right) \right] \tag{3.39}$$

$$\sigma_{yy} = \frac{K_I}{\sqrt{2\pi r}} \cos\left(\frac{\theta}{2}\right) \left[1 - \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{3\theta}{2}\right) \right] \tag{3.40}$$

$$\tau_{xy} = \frac{K_I}{\sqrt{2\pi r}} \cos\left(\frac{\theta}{2}\right) \sin\left(\frac{\theta}{2}\right) \cos\left(\frac{3\theta}{2}\right) \tag{3.41}$$

Figure 3.4 shows these solutions in front of the crack tip.

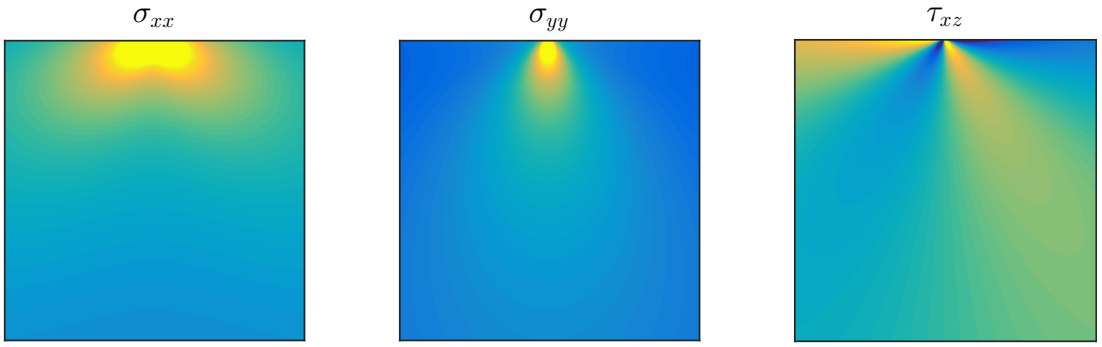


Figure 3.4: Near crack-tip stress for a plane crack along the y-axis. The crack tip is located at the top center of each plot.

Chapter 4

Molecular dynamics

4.1 Introduction

Molecular dynamics is a method for simulating systems of point particles. It is assumed that the particles behave classically; each particle obeys Newtons second law $\sum \mathbf{F} = m\ddot{\mathbf{r}}$. The forces are calculated using interaction potentials, and each particle can in principle interact with all other particles in the system:

$$\mathbf{F}_i = -\nabla U_i(\mathbf{r}^N) \tag{4.1}$$

Where $\mathbf{r}^N = \mathbf{r}_1, \ldots, \mathbf{r}_N$ are the positions of all the particles in the system. In the simple case of two-particle interactions of one type, the potential on each particle becomes:

$$\mathbf{F}_i = \sum_{j=1}^{N} -\frac{\partial U(\mathbf{r}_{ij})}{\partial \mathbf{r}_{ij}} \tag{4.2}$$

Given a potential U, this is in principle all that is needed to do molecular dynamics. However, many details arise when implementing specific potentials, and when the thermodynamic conditions for the simulations are to be controlled. In order to get results that are relevant to the real world, the potentials often have to be quite complicated, involving not only pairwise interactions, but also many-body interactions. While complicated potentials are just rules imposed within the framework presented above, details concerning controlling thermodynamic properties and boundary conditions will alter the fundamental equations given above.

There are also details concerning computational efficiency. To be able to simulate large systems over long periods of time – large and long is relative to molecular scales – the methods must ignore negligible parts of the potentials, but capture the important parts to sufficient accuracy.

4.2 Potentials

In molecular dynamics, it is natural to define two types of interactions, and therefore two types of potentials, based on how they relate groups of atoms. *Non-bonding interactions*

Table 4.1: Definitions of some of the reduced Lennard-Jones parameters.

Property	Definition
Length	$L^* = L/\sigma$
Energy	$E^* = E/\epsilon$
Density	$\rho^* = \rho\sigma^3$
Temperature	$T^* = k_B T/\epsilon$

depend on the spatial configurations of each possible set of particles of a specific kind, for example all triplets of Si-atoms or all HOO triplets (sorted that way). *Bonding interactions* depend on the spatial configuration of specific predefined sets of particles. The total potential is the sum of the contributions from these potentials:

$$U_{\text{tot}} = U_{\text{non-bonding}} + U_{\text{bonding}} \tag{4.3}$$

These definitions should not be confused with the distinction between covalent and non-covalent bonds in chemistry, even though it is common that bonding interactions in molecular dynamics represent covalent bonds.

4.2.1 Non-bonding potentials

I will use two non-bonding potentials in this thesis: The Lennard-Jones potential and the Coulomb potential. The Lennard-Jones potential is simple and widely used. For two particles separated by a distance r, the Lennard-Jones potential is:

$$U_{LJ} = 4\varepsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] \tag{4.4}$$

(Do not confuse σ and ϵ with stress and strain in elasticity). The first term represents Pauli repulsion, and the second represents van der Waals attraction. σ is a characteristic length that is close to the equilibrium distance between two particles. ε is the depth of the potential well – a measure of the strength of the interaction. The Lennard-Jones potential is computationally attractive because there is no need to calculate square roots of distances between particles. The Lennard-Jones potential is given in terms of reduced units. Since most studies of Lennard-Jones systems report results in these units, the equations defining some properties in reduced units are included in table 4.1.

The Coulomb potential – a potential for static charges – is used for charges particles, and is equivalent to Coulombs law:

$$U_e = k\frac{q_i q_j}{r_{ij}} \tag{4.5}$$

As an example of a simple three-particle non-bonding potential, I include the silicon potential by Stillinger & Weber [65]:

$$U_{SW} = \sum_i \sum_{j>i} \phi_2(r_{ij}) + \sum_i \sum_{j\neq i} \sum_{k>j} \phi_3(r_{ij}, r_{ik}, \theta_{ijk}) \tag{4.6}$$

$$\phi_2(r_{ij}) = A\varepsilon \left[B \left(\frac{\sigma}{r_{ij}} \right)^p - \left(\frac{\sigma}{r_{ij}} \right)^q \right] \exp\left(\frac{\sigma}{r_{ij} - a\sigma} \right) \tag{4.7}$$

$$\phi_3(r_{ij}, r_{ik}, \theta_{ijk}) = \lambda\varepsilon \left[\cos\theta_{ijk} - \cos\theta_0 \right]^2 \exp\left(\frac{\gamma\sigma}{r_{ij} - a\sigma} \right) \exp\left(\frac{\gamma\sigma}{r_{ik} - a\sigma} \right) \tag{4.8}$$

Note the exponential tails on the potential, resulting in the potential and its derivative going to zero when r_{ij} gets large. The functional form of this potential is quite flexible, and of the Stillinger–Weber potential has been used for example to model water, using one Stillinger-Weber particle per water molecule.

4.2.2 Bonding potentials

Bonded potentials only apply for a specific group of particles. A simple two-particle bonding potential is the harmonic bond potential:

$$U_{HB} = k(r - r_0)^2 \tag{4.9}$$

Where r_0 is the equilibrium distance between the particles and k is a constant of proportionality.

A simple three-particle bonding potential, which is often used in combination with the harmonic bond potential is the harmonic angle potential:

$$U_{HA} = k(\theta - \theta_0)^2 \tag{4.10}$$

These two latter potentials are common for creating small molecules that will not take part in chemical reactions, such as water in ice simulations. Sometimes these potential are taken to the limit of $k \to \infty$ creating rigid particles. The rigidity is then treated with dedicated algorithms.

4.3 Time integration

In order to solve the Newtonian equations of motion numerically, a discrete scheme is needed. For particles in conservative fields – fields in classical molecular dynamics are conservative – the velocity Verlet scheme is usually preferred [26, p.69].

4.3.1 Velocity Verlet

The velocity Verlet algorithm is:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \dot{\mathbf{r}}(t)\Delta t + \frac{1}{2}\frac{\mathbf{F}(t)}{m}\Delta t^2$$

$$\dot{\mathbf{r}}(t + \Delta t) = \dot{\mathbf{r}}(t) + \frac{1}{2}\frac{\mathbf{F}(t) + \mathbf{F}(t + \Delta t)}{m}\Delta t$$

The velocity Verlet integrator is symplectic. Being symplectic can be defined in a very strict mathematical way, but for the purposes of this work, it is sufficient to note that a symplectic integrator conserves a Hamiltonian that is only slightly perturbed relative to the Hamiltonian from which it tries to solve Hamilton's equations. This in turn means that in practice it conserves energy. Therefore, the velocity Verlet algorithm can be used to sample the microcanonical ensemble (NVE).

4.4 Simulation box

MD simulations can in principle be performed in infinite space, but it is usually best to confine the simulation to a specified region. Most commonly, the simulation is performed in a box with periodic boundary conditions (PBC). In PBC, a particle that leaves the simulation box reappears on the opposite side of the simulation box, keeping its velocity. At any point in time, the position of a particle can be measured in any *periodic image* by going through the periodic boundary as many times as wanted before actually measuring the position. The position of a particle in the simulations box, ignoring the periodic boundary condition is the *local cell* position of that particle, and the simulation box itself is thought of as the *local cell*. Note that a particle comes back to the local cell when it goes through the periodic boundary. The whole concept of the image cells is introduced to be able to handle interactions correctly. The most immediate complication introduced by PBC is that there are now several ways to go from one atom to another, depending on what periodic image positions of the particle positions are used. This has consequences for the force calculation. The usual solution is to apply the so-called *minimum image convention*, which means that each pair of particles is counted only once, and the distance vector is chosen to be the shortest one. In practice, the minimum image is calculated by first checking the non-periodic distance. Then, for each component of the difference vector, it is checked if the absolute value of that component can be reduced by adding or subtracting the simulation box length along that component. There are two main reasons why PBC in beneficial for simulations in confined regions: First, it is the simplest choice. Any other choice would imply either some sort of wall interaction or particles just disappearing. Secondly, it allows for sampling bulk properties of the system, since the system is quasi-infinite. This should, however, be done with caution, since there are well-known finite size effects of using PBC, such as the self-diffusion coefficient being lower for smaller simulation boxes (To be discussed in chapter 7).

4.5 Temperature

The instantaneous temperature T in molecular dynamics is usually defined using the equipartition theorem:

$$\langle E_k \rangle = \frac{f}{2} k_B T \tag{4.11}$$

Where f is the number of kinetic degrees of freedom, and T is the thermodynamic temperature defined by the differential coefficient of internal energy with respect to entropy, $\frac{1}{T} = \frac{\partial E}{\partial S}$. The thermodynamic temperature is not available, so an instantaneous temperature is defined. With point mass particles, using the instant value of the kinetic energy, the instantaneous temperature can be defined as:

$$\mathcal{T} \equiv \frac{2E_k}{f k_B} = \frac{1}{f k_B} \sum_{i=1}^{N} m_i v_i^2 \tag{4.12}$$

This definition takes into account all translational degrees of freedom in the system. Rotational degrees of freedom does not exist on a per-particle basis, since point particles

are considered. It the case of holonomic constraints, the reduction of number of degrees of freedom that these represent must be included. In addition, the boundary condition can reduce the number of degrees of freedom. The kinetic energy temperature measure is not the only possible temperature measure, and in principle, one could choose to measure the temperature only using non-kinetic energy contributions in the system.

When a temperature measure is defined, various schemes can be applied to control the temperature of a molecular dynamics simulation. Such schemes are called thermostats, and they all try to couple the MD system to an external heat bath, either to get the system to a specific temperature before running in the microcanonical ensemble (NVE), or to run a simulation at constant temperature and sample properties in the canonical ensemble (NVT). Some thermostats can be shown to make the system sample a known thermodynamic ensemble, whereas others do not correspond to any known ensemble. The Nosé–Hoover thermostat is an example of a thermostat that samples the canonical ensemble, while the Berendsen thermostat is an example of a thermostat that does not sample any know ensemble. The Nose-Hoover thermostat is also an example of a thermostat that explicitly changes the equations defining the system that is simulated, while for example the simpler Berendsen thermostat only implicitly change the equations by introducing an artificial velocity rescaling.

The simplest way to control the temperature is to rescale all velocities such that equation 4.12 yields the correct value for the temperature. However, that results in very unrealistic dynamics. The instantaneous temperature is not supposed to be constant, neither in the canonical ensemble nor the microcanonical.

4.5.1 Berendsen thermostat

A considerable improvement from the rescaling approach is the thermostat of Berendsen *et al.* [8]. The Berendsen thermostat introduces a timescale for the velocity rescaling by adding a temperature-dependent friction term to the equations of motion. The idea is to weakly couple the system to a heat bath by rescaling particle velocities to satisfy:

$$\frac{\mathrm{d}\mathcal{T}}{dt} = \frac{T_{\mathrm{bath}} - \mathcal{T}}{\tau} \tag{4.13}$$

Where T_{bath} is the temperature of the heat bath that is coupled to the MD-system to keep the temperature constant. τ is a characteristic time for the thermostat, and serves as a way to control the strength of the heat bath coupling. This step is applied just after velocity calculation in the time integration scheme. Since the portion of energy belonging to respectively potential and kinetic energy vary in time as particles move according to the equations of motion, the actual value of $\mathcal{T}$ will not develop according to 4.13 itself. The thermostat is just a mechanism that tries to push the system towards a kinetic energy corresponding to a given temperature T_{bath}.

The effective equation of motion resulting from using the Berendsen thermostat to rescale the velocities when using the velocity Verlet algorithm to integrate the motion is [36, p.128]:

$$\ddot{\mathbf{r}}_i(t) = \frac{1}{m_i}\mathbf{F}_i(t) - \frac{1}{2\tau}\left[\frac{T_{\mathrm{bath}}}{\mathcal{T}} - 1\right]\dot{\mathbf{r}}_i(t) \tag{4.14}$$

4.5.2 Nosé–Hoover thermostat

The Nosé–sHoover thermostat represents a more complicated coupling to a heat bath than the Berendsen thermostat. An MD system propagated using the Nosé–Hoover thermostat will sample the canonical ensemble. This presentation of the Nosé–Hoover thermostat follows the presentation of Hünenberger [36].

The Lagrangian[1] of an MD system with no temperature control (NVE).

$$\mathcal{L}(\mathbf{r}, \dot{\mathbf{r}}) = \frac{1}{2}\sum_{i=1}^{N} m_i \dot{\mathbf{r}}_i^2 - U(\mathbf{r}) \tag{4.15}$$

The basic idea of the Nosé-Hoover thermostat is to introduce an extra dynamic variable to the equations of motion, effectively creating an *extended* system, which is almost equal to the original system, which we will denote the *real* system. The Lagrangian of the extended system is chosen to be

$$\mathcal{L}_e(\tilde{\mathbf{r}}, \dot{\tilde{\mathbf{r}}}, \tilde{s}, \dot{\tilde{s}}) = \underbrace{\tfrac{1}{2}\textstyle\sum_{i=1}^{N} m_i \tilde{s}^2 \dot{\tilde{\mathbf{r}}}_i^2 - U(\tilde{\mathbf{r}})}_{\text{Microcanonical part}} \underbrace{+\tfrac{1}{2}Q\dot{\tilde{s}}^2 - gk_BT_{\text{bath}} \ln \tilde{s}}_{\text{Thermostat terms}} \tag{4.16}$$

where s is a new dynamic variable associated with a mass Q. g shall be chosen to be the number of degrees of freedom f if sampling is done in real time, and $f + 1$ if sampling is done in virtual time (explained below) for the thermostat to sample the canonical ensemble. The equation is now written in terms of new coordinates, which are the coordinates of the extended system. The new coordinates are given below, with extended system coordinates marked with $\tilde{}$:

$$\tilde{\mathbf{r}} = \mathbf{r}, \quad \dot{\tilde{\mathbf{r}}} = \tilde{s}\dot{\mathbf{r}}, \quad \tilde{s} = s, \quad \dot{\tilde{s}} = \tilde{s}^{-1}\dot{s} \tag{4.17}$$

The corresponding equations of motion are (in the extended system):

$$\ddot{\tilde{\mathbf{r}}}_i = m_i^{-1}\tilde{s}^{-2}\tilde{\mathbf{F}}_i - 2\tilde{s}^{-1}\dot{\tilde{s}}\dot{\tilde{\mathbf{r}}}_i \tag{4.18}$$

$$\ddot{\tilde{s}} = Q^{-1}\tilde{s}^{-1}\left(\sum_{i=1} m_i \tilde{s}^2 \dot{\tilde{\mathbf{r}}}_i - gk_BT_{\text{bath}}\right) \tag{4.19}$$

This is the thermostat of Nosé [54], and in the real system it samples the canonical ensemble. A problem with this thermostat, is that the velocity-scaling between the real and the extended system effectively leads to uneven time intervals in the real system. Running a simulation with the Nosé-thermostat is referred to as *virtual-time sampling*. Hoover [34] later overcame this problem, and showed that the equations of motion in the Nosé-algorithm could be equivalently formulated in real system coordinates (*real-time sampling*):

[1] The Lagrangian is the kinetic energy minus the potential energy: $\mathcal{L} = E_k - U$.

$$\ddot{\mathbf{r}}_i = m_i^{-1}\mathbf{F}_i - \gamma\dot{\mathbf{r}}_i \tag{4.20}$$

$$\dot{\gamma} = -k_b f Q^{-1}\mathcal{T}\left(\frac{g}{f}\frac{T_{\text{bath}}}{\mathcal{T}} - 1\right) \tag{4.21}$$

This is known as the Nosé-Hoover thermostat [34]. Note that this is still a formulation of the *extended* system, but it is formulated in *real* system coordinates. The *real* system – all $\ddot{\mathbf{r}}_i$'s – now sample the canonical ensemble in real system coordinates with an even timestep.

It is common to introduce a characteristic timescale for the Nosé-Hoover thermostat, since time is more intuitive than mass in MD simulations.

$$\tau_{NH} = (f k_B T_{\text{bath}})^{-\frac{1}{2}} Q^{\frac{1}{2}} \tag{4.22}$$

This quantity is usually an input parameter in MD-simulation packages providing a Nosé-Hoover thermostat.

Three types of temperature have now been introduced: Thermodynamic temperature T, instant temperature $\mathcal{T}$, and heat bath temperature T_{bath}. From here on, I will use the symbol T for all of them, unless two different temperatures appear in the same equation. It should always be clear from the context what T means. For example: A temperature referred from an experiment will always be a temperature measurement, which tries to measure T, while a temperature specified for a thermostatted molecular dynamics simulation will be T_{bath}.

4.6 Pressure and stress tensor

The instantaneous pressure in a general classical N-body system interacting with classical potentials can be expressed based on the virial equation for the pressure:

$$\mathcal{P} = \frac{N k_B \mathcal{T}}{V} + \frac{\langle W\rangle}{3V} \tag{4.23}$$

$\langle W\rangle$ is an ensemble average of the virial:

$$W(\mathbf{r}^N) \equiv -3V\frac{\mathrm{d}U(\mathbf{r}^N)}{\mathrm{d}V} \tag{4.24}$$

U is the potential energy of the system, and is only a function of the positions of the particles and the interaction potentials acting between them, since molecular dynamics are, except for temporary thermostat contributions, frictionless.

The virial is usually just derived for two-particle interactions, ignoring many-body interactions. But Thompson *et al.* [71] shows that the virial of a system with many-body interactions is not very different from that of the pair interactions, as long as the many-body interacting particles can be divided into groups. The virial of the local cell is then:

$$W(\mathbf{r}^N) = \sum_{k\in\mathbf{0}}\sum_{w=1}^{N_k}\mathbf{r}_w^k\cdot\mathbf{F}_w^k \tag{4.25}$$

Here k denotes groups, and w denotes particles in that group, so $\mathbf{r}_w^k$ is the wth particle in the kth group. $\sum_{k\in\mathbf{0}}$ is the sum over local-cell groups of the periodic system. There is a main particle in each group that is always defined to use its local cell position in interactions with the other particles in the group of which it is the main particle. If other particles' positions must be measured in a periodic image to obey the minimum image convention, then their position when calculating the contribution to the virial should be the local-cell position (even though this position is outside the simulation box).

The stress tensor is defined in a similar way to the pressure:

$$\mathbf{P}V = \langle \sum_{i=1}^{N} m_i \mathbf{v}_i \otimes \mathbf{v}_i \ + \mathbf{W}(\mathbf{r}^N) \rangle \tag{4.26}$$

The virial tensor, again found in Thompson *et al.* [71] is:

$$\mathbf{W}(\mathbf{r}^N) = \sum_{k\in\mathbf{0}} \sum_{w=1}^{N_k} \mathbf{r}_w^k \otimes \mathbf{F}_w^k \tag{4.27}$$

4.6.1 Barostats

Various schemes can be applied to achieve a desired pressure in a simulation. Usually, this is done by changing the simulation box volume. The basic idea of the box volume scaling schemes is the same as the velocity rescaling schemes for creating thermostats: The system is coupled to a pressure bath. Berendsen proposed a barostat rescaling the box volume in the same spirit as his thermostat rescaled velocities:

$$\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}t} = \frac{P_{\text{bath}} - \mathcal{P}}{\tau_P} \tag{4.28}$$

The box and all particle positions shall be rescaled to obey the above equation for the pressure change.

An MD system can be coupled to a pressure bath in a similar way as the thermal bath with the Nosé-Hoover thermostat. The real-time sampling version of this is due to Hoover [34], and is called the Nosé-Hoover barostat.

The Nosé-Hoover method was developed for isotropic pressure. Before publication of the extended-system methods, Parrinello & Rahman [56] created a method to control the full pressure tensor of an MD system, but without an extended system. Martyna *et al.* [48] used both of these methods to create a method with extended system coupling *and* control of the full pressure tensor. The details of these methods are out of the scope of this thesis.

4.7 Cutoffs and long-range corrections

Large-scale molecular dynamics simulations ($N \sim 10^6$) are unattainable if interactions between all particles are to be calculated. Molecular dynamics simulations with only pairwise interactions require $\mathcal{O}(N^2)$ calculations to be performed. To reduce the computational complexity, a cutoff radius r_{cut} is introduced: Only interactions between

particles that are closer to each other than the cutoff radius are calculated. Introducing a cutoff radius reduces the time complexity of an MD simulation from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$.

Most potentials fall off quickly, so that the error introduced by r_{cut} is negligible. For some potentials however, for instance the electrostatic potential, a cutoff introduces large errors, rendering simulation results useless. Luckily, there are techniques that handle long-range interactions without explicitly calculating the forces between every pair of particles. These methods are based on a method developed by Ewald [23] to calculate the energy of ionic crystals. I will briefly describe the idea of these methods – a thorough understanding and description is out of the scope of my project.

4.7.1 Ewald summation

For Coulomb point charges with periodic boundary conditions, the electrostatic energy is:

$$E_{\text{Coulomb}} = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\mathbf{n}\in\mathbb{Z}^3}\frac{q_i q_j}{|\mathbf{r}_{ij}+\mathbf{n}L|} \tag{4.29}$$

Where the inner sum over $\mathbf{n}$ is over all periodic images of the system. The inner sum omits $\mathbf{n} = (0,0,0)$ when $i = j$. The idea of the Ewald summation is to divide this sum into a short-range part E_{sr} and a long-rang part E_{lr}. The short-range contribution will be calculated in real space, and the long-range part will be calculated in Fourier space. The Fourier space calculation will deal with the Fourier transformed of the charge density:

$$\tilde{\rho}(\mathbf{k}) = \sum_{j=1}^{N} q_j e^{-i\mathbf{k}\cdot\mathbf{r}_j} \tag{4.30}$$

For the method to be effective, the variations in charge density in the part calculated in Fourier space must be sufficiently slowly varying so that its Fourier transform can be well represented by only a few $\mathbf{k}$-vectors.

There are several methods that calculate the Ewald sum. They differ for example in how they mesh the charge distribution for Fourier transforms. It is crucial for computational efficiency to be able to use the fast Fourier transform (FFT), so the mesh has to support FFT. In a paper concerning different methods to calculate Ewald sums, Deserno & Holm [21] states, in a thorough paper reviewing the accuracy of mesh routines for Ewald sums, that the particle-particle-particle-mesh routine (P^3M) by Hockney & Eastwood [32] is the "most accurate and versatile routine" for Ewald sums.

4.8 Rigid groups of particles

There are several ways to keep a group of atoms constrained as a rigid molecule. The goal of constraint algorithms is to obtain the same positions and velocities as if the equations of motion were integrated in coordinates incorporating the holonomic constraints, but without explicitly using these coordinates. The SHAKE algorithm by Ryckaert *et al.* [60] is the most common constraint algorithm. For the particular case of three rigid atoms, like a water molecule, the SETTLE algorithm [51] is an analytical solution when integrating numerically the equations of motion.

4.9 Measurements

I this section, I describe two transport properties (self-diffusion coefficient and viscosity) and one structural property (the radial distribution function), and explain how to measure them in MD.

4.9.1 Self-diffusion coefficient

Self-diffusion is the diffusion of a labeled particle of a species among other particles of the same species. Like in regular diffusion, the expected concentration of this particle in time will be governed by Fick's law. The self-diffusion coefficient can be measured with the Einstein relation:

$$D_E = \lim_{t\to\infty} \frac{1}{6}\frac{\mathrm{d}\langle|\mathbf{r}(t)-\mathbf{r}(0)|^2\rangle}{\mathrm{d}t} \tag{4.31}$$

Where 'E' means that D was estimated using the Einstein relation. It was shown in [78] that the diffusion coefficients in periodic simulation boxes depend on the system size, and goes like L^{-1}. Thus, a reference diffusivity D_0, corresponding to an infinite system and which can be compared with experimental results, can be found by extrapolation.

4.9.2 Viscosity

For calculating the viscosity, I use the Green-Kubo relation:

$$\eta_{GK} = \frac{V}{k_B T}\int_0^\infty \langle\sigma_{\alpha\beta}(t)\sigma_{\alpha\beta}(0)\rangle \,\mathrm{d}t \tag{4.32}$$

Where $\sigma_{\alpha\beta}$ are independent off-diagonal elements of the stress tensor of the system. These can be σ_{xy}, σ_{xz}, σ_{yz}, $(\sigma_{xx}-\sigma_{yy})/2$ and $(\sigma_{yy}-\sigma_{zz})/2$. The expectation value inside the integral assumes that a large number of systems are investigated. Through the ergodic hypothesis, we can transform the autocorrelation function into a time integral. Also, since infinite time series are not available in practice, I introduce finite bounds on the integrals.

$$\eta_{GK}(t,\tau) = \frac{V}{k_B T}\int_0^t \frac{1}{\tau}\int_0^\tau \sigma_{\alpha\beta}(\tau'+t')\sigma_{\alpha\beta}(\tau')\,\mathrm{d}\tau'\,\mathrm{d}t' \tag{4.33}$$

For the purpose of estimating viscosities, the autocorrelation function is only well sampled for $t \lll \tau$. Luckily, the autocorrelation function is essentially zero for times larger that some system-specific characteristic time, which is usually in the order of picoseconds, at least for water. That means $\eta_{GK}(t,\tau)$ can provide good estimates of the viscosity for molecular dynamics simulations that last for nanoseconds.

Following [78], it would also be possible to calculate the viscosity using the finite size effect on the self-diffusion constant.

4.9.3 Radial distribution function

The radial distribution function (RDF), $g(r)$, describes how the density of particles vary with the distance between particles. It can be defined as:

$$g(r) = 4\pi r^2 \rho(r)\ \mathrm{d}r \tag{4.34}$$

Where r is taken from a selected particle, and $\rho(r)$ is the number density of particles within the spherical shell $(r, r + \mathrm{d}r)$. In practice, the RDF is sampled using all particles as the origin in order to collect as much statistics as possible. The RDF is useful to identify crystal structures and for verification of simulations.

The simplest way to measure the RDF is to loop over all pairs of particles, measure the distance between them and put the distances in bins.

Part II

Numerical models

Chapter 5

A model for methane hydrates

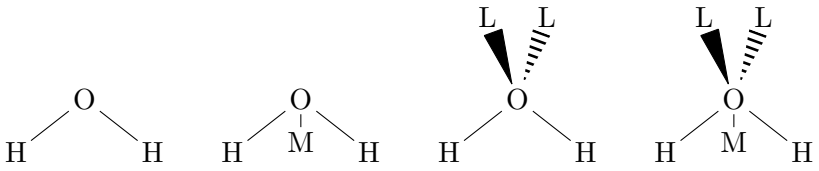
With the theoretical framework of molecular dynamics in place, I go on to introduce specific potentials, and choose which one to use to model cracks in methane hydrates. This chapter bridges the theory presentation to my work.

Some force fields are developed with one specific application in mind, while others give a general functional form that can be parametrized for a wide range of applications. Examples of the latter are the AMBER force field [19] and the OPLS force field [40]. The OPLS force field will be explained in more detail. It is also common to distinguish these force fields from the water potentials that are used along with them. Since water is complicated to model, it is best treated separately. In this chapter, I will describe the TIP4P water model and briefly mention the SPC water model.

5.1 Criteria for a sufficient model

I will model fracture of methane hydrates. If the methane hydrates are pure, i.e. no sediments nearby, it is reasonable to assume that no chemical reactions will take place. Conversely, if the hydrates are to be modeled near for example a silicon oxide surface, chemical reactions can be important. Since chemical reactions are not expected to occur in pure hydrates, bonding potentials can be used for covalent bonds. The transferable interpotential models (TIP-models) are popular models that use bonding potentials for the OH-bonds. For applications on fracture, it is important that mechanical properties are reasonably represented.

Since this is the first time I choose a molecular dynamics potential, I will use a model that other people have shown to work. Specifically, I choose the model that has been shown to spontaneously nucleate methane hydrates in molecular dynamics simulations [76]. Thus, the strategy is to use a model that successfully reproduced some feature of methane hydrates to study another feature. This means that the model cannot be immediately trusted, and this study is just as much a test of whether the model I choose is suitable for studying fracture of methane hydrates as an actual study of fracture.



3-site model 4-site model 5-site model 6-site model

Figure 5.1: Different kinds of molecular water models.

5.2 Water models

Molecular dynamics water models most commonly contain either 3 or 4 interaction sites. Models with 5 or 6 sites exist, but they are more computationally expensive and less commonly in use. The 5- and 6-site models were developed later than the 3- and 4-site models, which might also be a reason why they are less popular. A clear difference between 5- and 6-site models and the ones with fewer sites, are the two L-sites – two charged sites near the oxygen site – which gives more structure to the energy surface of the *water dimer* [44]. A selection of common site configurations are given in figure 5.1.

All the water models illustrated in figure 5.1 have their mass placed on the oxygen and hydrogen sites. Additionally, there is a positive charge on each hydrogen atom. The negative charge associated with the oxygen atom is either put on the atom itself (3-site model) or in its entirety moved to one, two or three massless sites (4-, 5- and 6-site models) called M-site and/or L-sites. These massless sites distribute the force acting on them among the massive particles when the equations of motion are integrated.

For simple water models (which are widely used), there are Coulomb interactions between charged sites, and Lennard–Jones interactions between the oxygens. Some models also include a Lennard–Jones interaction between the hydrogen sites, but that is less common.

5.2.1 SPC water model

The SPC water model [9] is a 3-site model with a different spatial configuration than the experimental configuration of the nuclei of the water molecule. Rather than using the experimental HOH angle of 104.52°, it uses the ideal tetrahedral angle of 109.47°. The sites retain their experimental masses. The interactions in this model is a Lennard–Jones interaction between oxygens, and a Coulomb force between both OH-pairs and HH-pairs, provided that they belong to different molecules. The water molecule itself is usually kept rigid, but there exist parameters for a flexible version of the potential.

5.2.2 TIP4P water model

The TIP4P water model was introduced by Jorgensen *et al.* [41] in 1983. Since then, the model has gone through several parameterizations. The TIP4P/2005 parameter set has been found to give the best overall performance in water simulations, but there also exist a parametrization specifically made for the solid phases of water: TIP4P/ICE, parametrized by Abascal *et al.* [1]. TIP4P was used by Matsumoto *et al.* [49] in

the first successful simulation of nucleation of ice crystals in molecular dynamics, and TIP4P/ICE was used by Walsh *et al.* [76] in the first successful simulation of spontaneous methane hydrate nucleation.

TIP4P is a rigid water model with 4 sites for each water molecule: One O-site, two H-sites, and a site commonly referred to as the M-site. This corresponds to the second model from the left in figure 5.1. The M-site is supposed to slightly move the oxygen charge, and it is situated on the bisector of the H-O-H angle. The parameters for the spatial configuration, as well as the masses of hydrogen and oxygen, are taken from experimental observations, and were listed in table 2.2. The HOH-angle and the OH-bonds are rigid. The electrostatic interactions are coulombic, and the oxygen–oxygen interaction is a Lennard–Jones interaction. The parameters are given in table 5.1. For reference, parameters for the original TIP4P model and the 2005 reparametrization are given in table 5.2.

Table 5.1: Parameters for TIP4P-ICE model as in [1].

Description	Symbol	Value
Lennard-Jones energy	ε/k	100.5 K
Lennard-Jones characteristic distance	σ_{OO}	3.155 Å
Distance O-site to M-site along bisector	d_{OM}	0.157 Å
Hydrogen charge	q_H	$0.5676\,e$
Oxygen charge	q_O	$2q_H$

Equations for the TIP4P interactions To be explicit about the functional forms of the TIP4P model, I give them below:

$$E_{LJ} = 4\varepsilon\left[\left(\frac{\sigma}{r_{OO}}\right)^{12} - \left(\frac{\sigma}{r_{OO}}\right)^{6}\right] \tag{5.1}$$

$$E_e = \frac{e^2}{4\pi\varepsilon_0}\sum_{a,b}\frac{q_a q_b}{r_{ab}} \tag{5.2}$$

The Lennard–Jones interaction is between oxygens, and the electrostatic interaction is between all charged sites of different molecules (but not between sites belonging to the same molecule).

5.3 Optimized Potentials for Liquid Simulations (OPLS)

5.3.1 Potential model

The OPLS force field has the following contributions to the potential energy:

$$E(\mathbf{r}^N) = E_{\text{bond}} + E_{\text{angle}} + E_{\text{torsion}} + E_{\text{nonbonded}} \tag{5.3}$$

Table 5.2: Parameters for the different parameterizations in the TIP4P family.

Description	Symbol	TIP4P	TIP4P/2005
Lennard-Jones energy	ε/k	78.0 K	93.2 K
Lennard-Jones characteristic distance	σ_{OO}	3.154 Å	3.1589 Å
Distance O-site to M-site along bisector	d_{OM}	0.150 Å	0.1546 Å
Hydrogen charge	q_H	0.520 e	0.5564 e
Oxygen charge	q_O	$2q_H$	$2q_H$

Where the individual energy contributions are defined as follows:

$$E_{\text{bond}} = \sum_{\text{bonds}} K_r(r - r_0)^2 \tag{5.4}$$

$$E_{\text{angle}} = \sum_{\text{angles}} K_\theta(\theta - \theta_0)^2 \tag{5.5}$$

$$E_{\text{torsion}} = \sum_{\text{dihedrals}} \frac{1}{2}K_1(1 + \cos\phi) + \frac{1}{2}K_2(1 - \cos 2\phi) + \frac{1}{2}K_3(1 + \cos 3\phi) + \frac{1}{2}K_4(1 - \cos 4\phi) \tag{5.6}$$

$$E_{\text{nonbonded}} = \sum_{i>j} f_{ij}\left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{C_{ij}}{r_{ij}^{6}} + \frac{q_i q_j e^2}{4\pi\epsilon_0 r_{ij}}\right) \tag{5.7}$$

For the torsion contribution, ϕ is the dihedral angle (the angle between two planes) defined by three vectors connecting four atoms. The torsion contribution given here is slightly simplified compared to the original OPLS force field, but it *is* the version that is implemented in LAMMPS.

The non-bonded interaction is a Lennard–Jones plus a Coulombic potential. The prefactor f is the so-called *fudge factor*, and is 0.5 for particles that are three bonds apart. Closer than 3 three bonds it is zero, and otherwise it is 1.

All interaction-sites are centered on the atoms, i.e. there are no massless sites like in the water models.

5.3.2 OPLS United-atom methane

The methane hydrate model of Walsh *et al.* [76] (which I am going to use) uses one of the simplest OPLS-models for methane: OPLS United-atom (OPLS-UA). United atom methane is the united atom model for methane, and is effectively a single Lennard-Jones interaction site for each methane molecule.

$$E_{MM} = 4\varepsilon\left[\left(\frac{\sigma}{r_{ij}}\right)^{12} - \left(\frac{\sigma}{r_{ij}}\right)^{6}\right] \tag{5.8}$$

A parameter set for this methane representation can for example be taken from [47], and are listed in 5.3.

Table 5.3: Lennard-Jones parameters for united atom methane.

Description	Symbol	Value
Lennard-Jones energy	ε/k_B	147.9 K
Lennard-Jones characteristic distance	σ_{MM}	3.73 Å

5.4 Combining particles of different species in a model

Having different potentials for different aggregate particles (such as TIP4P/ICE), we need a way to combine them in a model containing both species. For electrostatic interactions, the combination rules are straightforward, as they follow from Coulombs law. For Lennard-Jones interactions, combination rules do not appear automatically, and in principle one has to fit parameters for each unique pair of particles. Fortunately, formulas that work reasonably well without going into an extensive parameter fitting exercise have already been developed.

One simple way, which is the one that is used in Walsh *et al.* [76], is to use the following rules from two sets of parameters (ε_{ii}, σ_{ii}) and (ε_{jj}, σ_{jj}) for interactions between two particles of the same species i or j, on to a set of parameters (ε_{ij}, σ_{ij}) for interactions between one particle of each species:

$$\varepsilon_{ij} = \sqrt{\varepsilon_{ii}\varepsilon_{jj}} \tag{5.9}$$

$$\sigma_{ij} = \frac{1}{2}(\sigma_{ii} + \sigma_{jj}) \tag{5.10}$$

These are called Lorentz–Berthelot combination rules.

5.5 TIP4P-ICE + OPLS-UA methane

The necessary ingredients for a potential model for methane hydrates are now presented: A water model, a methane model, and a way to make the water and the methane interact with each other. The model is illustrated in figure 5.2. This model is essentially a description of the total energy of a system consisting of particles with known positions and momenta. The next step is to decide on what kinds of systems to model, i.e. what are the initial positions going to be, and that are the thermodynamic conditions going to be during a simulation?

5.6 A system for studying fracture of methane hydrates

It was suggested by Ning *et al.* [53] to find basic mechanical properties like Young's modulus and Poisson's ratio and the fracture toughness of methane hydrates using molecular dynamics. Particularly, they propose to study methane hydrates with a dislocation subjected to tensile stress. Since I have not found any studies on the tensile strength of methane hydrates using molecular dynamics, I choose to model a system close to the simplest case in fracture mechanics: A rectangular prismatic piece of

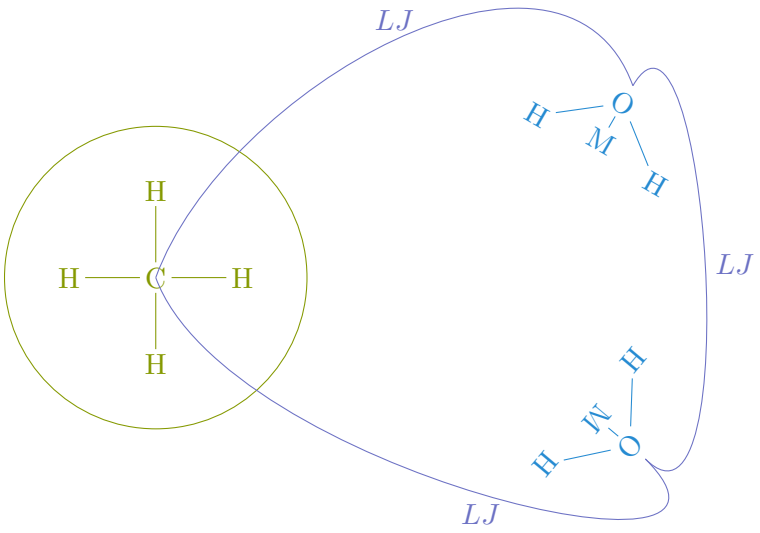


Figure 5.2: Lennard-jones interactions between methane and water in the TIP4P + OPLS-UA methane model. The hydrogens on the methane are not present in the model. The circle around the methane indicates that it is *one* particle. Coulombic interactions are not indicated, but they act between the H- and M-sites of the two water molecules.

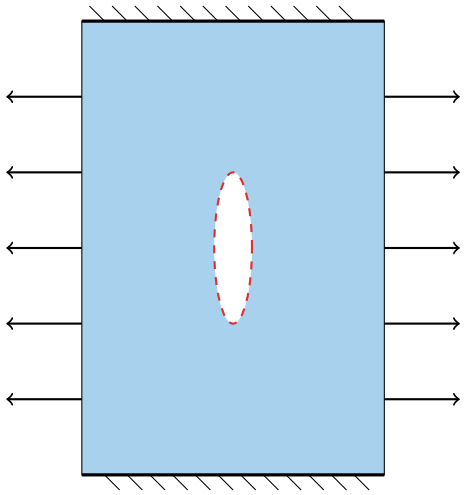


Figure 5.3: Illustration of the model system. The blue area is initially a pure sI methane hydrate crystal, and the area inside the red dashed ellipse an empty crack. The system will be subjected to tensile strain in the directions of the arrows. The other edges of the simulation box will be kept fixed.

methane hydrate with an artificial elliptical prismatic crack spanning its z-direction. This system will be subjected to tensile strain normal to the crack plane. Apart from the crack that is carved out, the hydrate lattice will be fully occupied with methane molecules. Figure 5.3 illustrates this system. Studies of partly occupied lattices are left for future work. Imposing shear stress would also be of immediate interest, but unfortunately, the current distribution of LAMMPS do not support a non-orthogonal simulation box if the TIP4P water model is going to be used with long-range forces.

Chapter 6

Simulation and analysis tools

The molecular dynamics workflow can be usefully divided into the following steps:

1. Create initial condition – set initial positions and velocities for all particles and define bonding interactions.
2. Run molecular dynamics with a set of parameters.
3. Post-process.
4. Analyze results.

Usually, step 3 will provide experience that influence how steps 1 and 2 are performed during the next iteration of the workflow. It is important to have tools in place to accommodate this workflow. Most importantly, the tools have to be directly compatible when iterating $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$. Otherwise, it is impossible to work efficiently. Second, it is convenient with compatibility $2 \rightarrow 1$ if a simulation needs to be run further in time. Third, it is convenient with compatibility $4 \rightarrow 1$ and $4 \rightarrow 2$, but the nature of such a compatibility is not as clear as the others, and is probably a matter of insight of the user rather than a technical compatibility.

The most fundamental choice concerning tools is whether to program the molecular dynamics simulator myself or to use a program made by someone else. I have implemented my own simple molecular dynamics program before, and that experience tells me to use a package. It is quite easy to implement a simple molecular dynamics solver for Lennard-Jones particles in NVE and Berendsen NVT, but more complex potentials and barostatic ensembles are more challenging to implement and test. I used the Wikipedia overview of molecular dynamics simulators as a first guide for choosing a simulator, and then moved on to scanning scientific papers on topics adjacent to my project. From that, I considered that either LAMMPS or GROMACS would be the best choice of simulator. After a brief look in the documentation of the two packages, I chose LAMMPS, due to its user-friendliness. Another advantage of LAMMPS is that it is easy to extend with own code if the package itself doesn't contain sufficient functionality. That means LAMMPS covers step 2 of the workflow.

Step 1 can also be partially covered by LAMMPS. First, LAMMPS have several built-in features to create initial conditions. Secondly, it is possible to code extensions

to LAMMPS to create and modify initial conditions. I have also used a tool called Moltemplate to create LAMMPS-compatible data files describing a methane hydrate.

I found it useful to distinguish between step 3 and step 4, even though the difference is not too obvious. One way of explicitly separating them is by the tools used in each process. Post-processing is done by another tool than the tool that presents the data for analysis.

The difference between steps 3 and 4 is not as clear as the difference between the other steps. I define it the following way. Post-processing is the processing of data prior to visualization. Analysis is the act of visualizing data that is already processed, and using that data to draw conclusions from the simulations. I will use Visual Molecular Dynamics (VMD) [35] for visualization, and custom tools I develop myself in MATLAB, PYTHON and C++ for data analysis.

In the following sections of this chapter, I give an introduction to LAMMPS that contains the features that will be used later on, and then describe post-processing tools that I have developed.

6.1 LAMMPS

LAMMPS [57], Large-scale Atomic/Molecular Massively Parallel Simulator, is a tool for (possibly) large particle-style simulations - mainly molecular dynamics. LAMMPS is a massive open source project, and its main features are developed and maintained at Sandia National Laboratories in the United States.

6.1.1 Input files

LAMMPS input files are a set of commands to be executed. Some commands are only valid in the right order. The structure of such a file is:

1. **Initialization:** Define the units (e.g. SI or `real`) of the simulation. Describe the simulation box. Define the interactions that will go into the simulation.

2. **Atom definition:** Set the initial positions (and optionally velocities) of the particles in the simulation. This can be done either from an input file or by generating positions according to some supported algorithm. The box boundaries are set in this section.

3. **Settings:** Define the parameters of the particle interactions. Create output objects for thermodynamic data and particle trajectories etc. Optionally create particle velocities. Create the integrator (which can contain a thermostat).

4. **Run:** Run the simulation.

A small example of how a LAMMPS file for running a simulation with TIP4P/Ice water and united-atom methane is shown below. The initial positions are expected to come from a LAMMPS data file (explained later).

```
1 # ----------------- Init Section -----------------
```

```
2  units       real
3  dimension     3
4  boundary    p p p
5  atom_style    full
6  pair_style    lj/cut/tip4p/long 1 2 1 1 0.1577 10.0
7  kspace_style  pppm/tip4p 1.0e-4
8  bond_style    harmonic
9  angle_style   harmonic
10 pair_modify   mix arithmetic
11
12 # ----------------- Atom Definition Section -----------------
13 read_data "water_methane_test.data"
14
15 # ----------------- Settings Section -----------------
16 pair_coeff  1   1   0.21084   3.1668
17 pair_coeff  2   2 0.0     0.0
18 pair_coeff  1   2   0.0     0.0
19 bond_coeff  1   0.0 0.9572
20 angle_coeff 1   0.0 104.52
21 pair_coeff  3   3    0.29391  3.73
22 pair_coeff  1 3   0.18  3.44
23 pair_coeff  2 3   0.0   0.0
24
25 group water type 1 2
26 group methane type 3
27 fix fShakebond  water   shake   0.0001 100 0 a 1 b 1
28
29 dump myDump all custom 5 trajectory.lammpstrj id element x y z vx
       vy vz
30
31 velocity all create 300.0 32352 rot yes mom yes dist gaussian
32 fix fxnpt all npt temp 300.0 300.0 100.0 x 400.0 400.0 100.0
33
34 # ----------------- Run Section -----------------
35 run   5000
```

First, units are chosen. The meaning of `real` units is best checked in the documentation [61]. The choice of units has consequences for all physical properties that are to be set, such as temperature and interaction energies. On line 3, the simulation is set to be three-dimensional, and on line 4, all three boundaries are set to be periodic. `atom_style` limits what kinds of interactions are possible to use for a simulation using the particular input file. Full flexibility is obtained here by setting `atom_style full`. `pair_style` sets the potential for the non-bonding interactions, as described in 4.2.1. Many types of `pair_style` potentials are available, and they can also be combined – but with caution – using the concept of hybrid pair styles. The `kspace_style` sets the long-range solver. Here it is set to P^3M, which is the only available long range solver in LAMMPS for the TIP4P water model. Then the bonding potential models are set to be harmonic (line 8-9). Lorentz-Berthelot combination rules are imposed with the command `mix arithmetic`. The `read_data` command reads a LAMMPS data file to import initial positions and bonding specification for all particles in the simulation. On lines 16-23, the coefficients for all the interactions are set. The first two arguments are the types of particles for which the coefficient shall be utilized, and

the rest are the actual coefficients in an order dictated by the documentation of that specific potential. For the potentials here, the first coefficient is an energy, and the second either a distance or an angle. Then, two groups of particles are defined for convenience, and on line 27, the holonomic constraints on the water molecules are set to be handled by the SHAKE-algorithm. `a 1 b 1` means that bonds of type 1 and angles of type 1 are to be constrained. The existence of these types of bonds and angles are set in the data file. Then, on line 29, the `dump` command sets how the positions and velocities of particles are to be written to file. Here, all particles are set to be dumped to the file `trajectory.lammpstrj` every fifth time-step. Then all particles are given an initial velocity from the Boltzmann distribution (Gaussian in each coordinate), and in line 32, the simulation is chosen to be run in the isothermic-isobaric ensemble (NPT). The standard thermostat for NPT in LAMMPS is Nosé–Hoover both for the temperature and the pressure. The actual equations that are used are the ones from Shinoda *et al.* [62]. The `run` command on the final line sets the number of time-steps to be simulated.

Data files

It is usually a good idea to have all atom input information in a file separate from the input file, and read it with the data_read command. An atom data file shall contain positions and velocities for all particles. In addition, if some atoms are parts of specific molecules, this is declared. All bonds and angles (and dihedrals and so on ..) that are meant to support interactions between specific particles must be declared. In addition, it is possible to insert coefficients for the different interactions. However, it is not possible to declare atom_style, pair_style or any other style in the data files. This has to be done in the input file, and it is important that the number of coefficients in the data file is compatible with the interaction potential chosen in the input file.

Below are some extractions from an input file for bulk water with bonded O-H interactions, and angular H-O-H interactions. I have used this file to simulate TIP4P water, by keeping the bonds and angles rigid with the shake-algorithm.

```
1  LAMMPS data file via write_data, version 11 Nov 2013, timestep =
       15000
2
3  10125 atoms
4  2 atom types
5  6750 bonds
6  1 bond types
7  3375 angles
8  1 angle types
9
10 -1.2998053548485800e-01 4.6629980535484322e+01 xlo xhi
11 -1.2998053548485800e-01 4.6629980535484322e+01 ylo yhi
12 -1.2998053548485800e-01 4.6629980535484322e+01 zlo zhi
13
14 Masses
15
16 1 1.00794
17 2 15.9994
```

```
18
19 PairIJ Coeffs
20
21 1 1 0 0 10
22 1 2 0 0 10
23 2 2 0.16275 3.16435 10
24
25 Bond Coeffs
26
27 1 0 0.9572
28
29 Angle Coeffs
30
31 1 0 104.52
32
33 Atoms
34
35 721 241 2 -1.0484400000000000e+00 -6.2221407531032169e-02
       1.5159166496011423e+00 5.3069458514672112e+00 0 0 0
36 1396 466 2 -1.0484400000000000e+00 4.0240333910499935e+00
       4.2356159167994916e+00 1.7598269969646658e+00 0 0 0
37 1398 466 1 5.2422000000000002e-01 4.7736723529825049e+00
       3.6734646891601108e+00 1.9555433512796152e+00 0 0 0
38 718 240 2 -1.0484400000000000e+00 1.6591102071344288e+00
       1.3108233273533909e-01 3.7368290353832792e+00 1 0 0
39 2115 705 1 5.2422000000000002e-01 2.6288644528128606e+00
       1.8244264320027685e+00 3.9483095929557672e+00 1 0 0
40 .
41 .
42 .
43
44 Velocities
45
46 721 -5.0354515315883175e-04 5.3167587742950842e-03
       3.3571203451087726e-03
47 1396 2.9628128063741773e-04 -3.6999987020118080e-03 -
       6.2672614595795321e-03
48 1398 1.9834410211762494e-03 -4.5261896434727186e-04 -
       3.5219356644226700e-03
49 718 6.4570329589514993e-04 -1.2753082693010263e-03
       5.5441917429848477e-03
50 2115 1.9812407347396029e-03 -1.4861789372548064e-03
       2.0816584325493676e-02
51 143 -7.7275724561505395e-03 1.4979358537740204e-03
       1.9557938914574084e-02
52 720 -8.5445971109074596e-03 -1.7755485312369985e-02
       1.4432539996760280e-02
53 2114 8.2504630257354671e-03 -1.5862538887645005e-02
       1.1754578049979690e-02
54 722 1.8121516439040641e-02 -5.4646869701048756e-03
       2.3672874162716955e-03
55 767 -7.7201655772992407e-03 -5.5203125867209482e-03 -
       1.2033511911111539e-02
56
57 .
58 .
```

```
59 .
60
61 Bonds
62
63 1 1 721 722
64 2 1 721 723
65 3 1 1396 1397
66 4 1 1396 1398
67 5 1 718 719
68 6 1 718 720
69
70 .
71 .
72 .
73
74 Angles
75
76 1 1 722 721 723
77 2 1 1397 1396 1398
78 3 1 719 718 720
79 4 1 9452 9451 9453
80 5 1 767 766 768
81 6 1 2114 2113 2115
82 7 1 53 52 54
83 8 1 143 142 144
```

The first lines are ignored by LAMMPS, so the first line to be interpreted is line 3. The first section, line 3-8, contains information about what is supposed to be contained in this file. The next section defines the bounds on the simulation domain. Then comes a definition of the masses for the two atom types that were declared in the first section. In the section called PairIJ Coeffs, parameters for pair_style interactions are set. The last line in this section tells that pair_style interactions between particles of type 2 and 2 in this simulations have parameters 0.16275, 3.1643 and 10. The first two parameters in the 1 1 and 1 2 interactions have value 0, which tells us that these do not interact pair_style. The bond coefficient section declares a bond with id 1. If there were to be additional types of bonded interactions, they would be numbered from 2 and so on. If the bond_style is set to harmonic, the parameters for the first bond tells that the equilibrium distance in this harmonic bond is 0.9572, but that the coefficient for the harmonic force is 0. The reason for the coefficient to be zero in this specific case, is that the bond is to be rigid, and rigidity is handled later, by a fix-command. The angle coeffs section follows the same pattern as the bond coeffs. The contents of the Atoms section can vary, depending on the atom_style used in the simulation. This setup is for atom_style full. Line 35 tells that atom number 721 belongs to molecule number 241. It is of type 2 (oxygen in this case), and has charge -1.04844. Then comes xyz positions, and an nx ny nz counter telling how many times the atom has passed each periodic wall. Velocities are self-explanatory. The bonds section lists all the bonds that are defined in the simulation. In this case it is O-H-bonds in water. line 63 tells that the first such bond is of type 1 (as defined in bond coeffs), and that it creates a bonded interaction between atoms 721 and 722. The angles section is similar, but it takes three atoms to make an angle, with the vertex in the middle.

This type of declaration is quite powerful. We have only defined that we have some water molecules, and that they support bonded and angular interactions, as well as pair_style interactions. That means we can use any potential (supported by LAMMPS) that have these restrictions. The only thing that has to be changed, are the coefficients for the different interactions. If one wishes to change potentials regularly, it can be a good idea to put coefficients in a file together with the declaration of potentials, using the <pair, bond, angle>_coeff commands.

6.2 Extending LAMMPS to cut elliptical prisms

LAMMPS was designed to be easily extendable, and in my opinion that turns out to be true. How to create extensions is explained in the section *Modifying & extending LAMMPS* in the LAMMPS documentation [61]. I have made *one* extension to LAMMPS, to be able to cut elliptical prisms. LAMMPS already contain features to remove atoms if they are defined to be in a group, so I only need to implement a way to define elliptical prismatic regions. The main contents of this extension is a function that determines whether a particle with a position (x, y, z) is inside or outside the region:

```
int RegEprism::inside(double x, double y, double z)
{
  double dx = x-xmid;
  double dy = y-ymid;
  return ( (dx/a)*(dx/a)+(dy/b)*(dy/b) ) < 1.0;
}
```

The whole extension is implemented as a class that inherits from the LAMMPS `class Region`. From the class definition of class region, it can be seen that its subclasses are required to implement three functions:

```
virtual int inside(double, double, double) = 0;
virtual int surface_interior(double *, double) = 0;
virtual int surface_exterior(double *, double) = 0;
```

I am not going to use the surface functions, so I only need to implement `inside(...)`. I leave the other two as dummy functions returning `false`, which means this region will be incompatible with other LAMMPS functionality if it requires knowledge of the surface properties of the region.

6.2.1 Usage

The usage of this extension is fairly straightforward. After compiling LAMMPS with the `eprism` extension, an elliptical prismatic region can be defined by the following command:

```
region <name> eprism <center x> <center y> <length x> <length y>
```

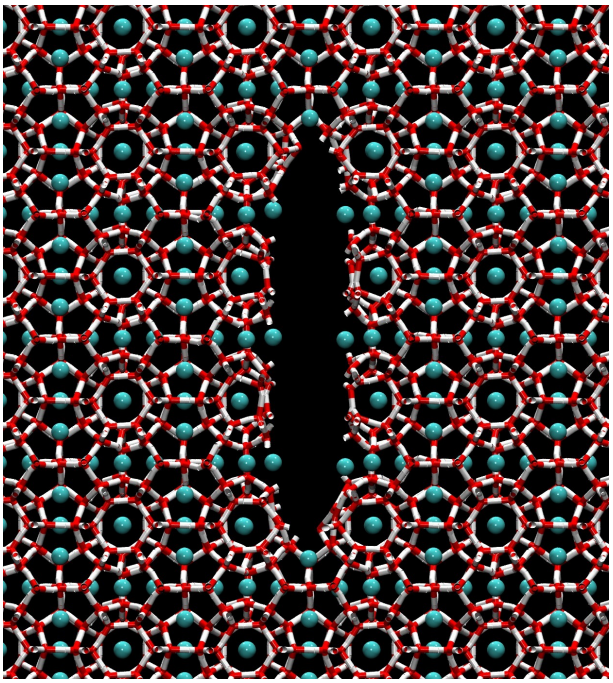


Figure 6.1: An elliptic crack carved out using `region eprism`

6.3 Crack tracer

Measuring the area of a crack surface is far from a trivial task. Due to the fractal-like interface between a molecular surface and the "void", at least one adjustable parameters must be included in a method for calculating the surface are of a crack in a molecular system. One possibility is to measure the solvent-accessible area, which has *one* free parameter: The radius of a trial particle rolling over the molecular surface.

6.3.1 Algorithm

I have implemented a Monte-Carlo crack tracer to measure the solvent-accessible area of a system of particles. The basic idea is that every point in space is defined as either part of the *void* or part of the *wall* based on the following definition: A void point is a point for which there exist no particles within a distance r_p. Points that are not part of the void are parts of the wall. The Monte-Carlo method I employ is as follows: N line segments of length Δl and uniformly random orientation and position in space are drawn. A surface point is detected if one of the edges is in the void while the other is in the wall. The line segments must be sufficiently short to only cross the void–wall interface once. The solvent-accessible surface area of the system can then be calculated with the following formula:

$$A_{ss}(r_p) = 2V\frac{n_s(r_p)}{N\Delta l} \tag{6.1}$$

Which is a simplified version of the formula used in Bhattacharya *et al.* [11]. n_s is the number of surface points detected among N line segments. V is the volume of the system. The length Δl is the only numerical resolution parameter for this measure, and has to be small compared to r_p in order to capture the surface structure. Note that one could in principle draw line segments of different lengths. The denominator just has to be the sum of the length of all line segments. The formula can actually be written

$$A_{ss}(r_p) = 2V\frac{n_s(r_p)}{L} \tag{6.2}$$

n_s is now the total number of void–wall crossings of an arbitrary set of straight line segments of total length L. Each line is now allowed to cross the void–wall interface arbitrarily many times, provided that the total number is added to n_s.

Figure 6.2 shows the method in action, with the blue area as the wall and the white area as the void.

6.3.2 Implementation

Cell lists The system is divided into a maximum number of equally sized cells such that that all cell dimensions l_x, l_y, l_z are greater than r_p. When a point is checked for whether it is part of the void or the wall, only the distance to particles belonging to the same cell and the directly and diagonally adjacent cells need to be checked. Using cell lists rather that looping over all particles in the system gives huge efficiency gains.

Periodic boundaries Periodic boundaries are implemented using cell lists, which means that the system must be at least $2 \times 2 \times 2$ cells – each dimension in the system must be at least $2r_p$.

Scaling efficiency If we assume that the required number of samples to estimate the surface area of a system is proportional to the number of particles in that system, the problem scales as $\mathcal{O}(N^2)$. With cell lists the problem scales like $\mathcal{O}(N)$.

6.3.3 Choice of parameters

As mentioned, there are two parameters to be chosen when applying this technique. First, r_p has to be chosen to define what surface is to be measured. Different r_p correspond to different solvent accessible surfaces. The resolution at which this surface is to be measured shall be chosen. A low value of Δl will capture details of the surface, and Δl should be significantly smaller than the characteristic size of the surface roughness. The cost of a small Δl is bad statistics. The shorter the line segments, the smaller the portion of segments that cross the void-wall boundary. Figure 6.3 shows the calculated crack surface area as a function of Δl and r_p for a crack in a sI hydrate during crack propagation.

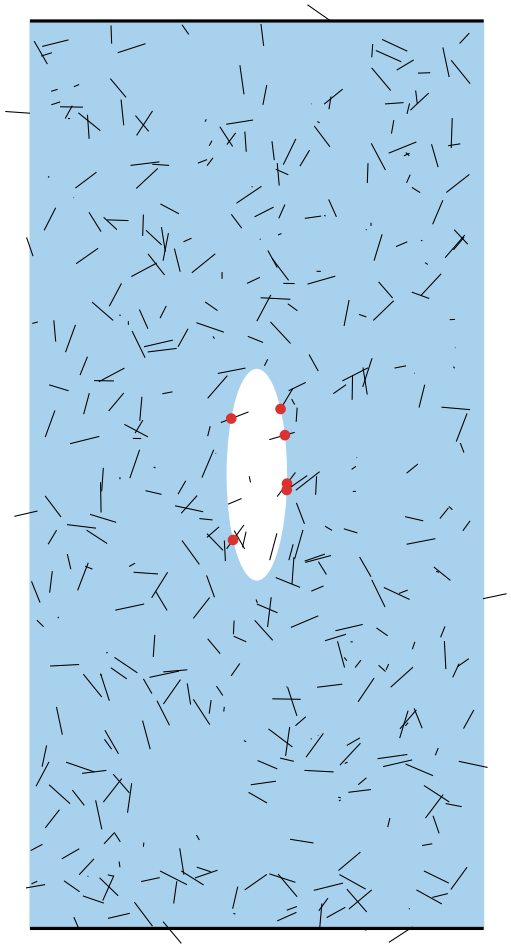


Figure 6.2: Illustration of the crack tracer method. Here, 400 line segments of length 0.4 cm are drawn. The line segments appear to have different lengths since they also have a component perpendicular to the paper plane. Red dots indicate the intersection of a line segment with the crack surface. If thinking in two dimensions, the area of this system, $6\,\mathrm{cm} \times 12\,\mathrm{cm}$, takes the role of the volume, and the surface area becomes a length. The calculated surface length is then: $2 \times 72\,\mathrm{cm}^2 \frac{6}{400 \times 0.3\,\mathrm{cm}} = 5.4\,\mathrm{cm}$. The true circumference of the crack in this example is around 6.5 cm.

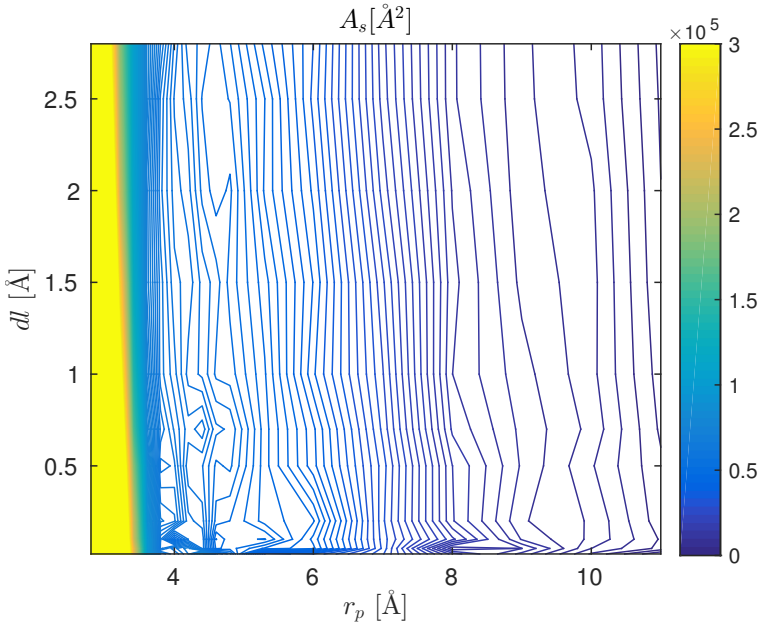


Figure 6.3: Calculated crack surface area as a function of Δl and r_p for a crack propagating in a system of $24 \times 24 \times 12$ sI unit cells. The high values corresponding to low values of r_p are because the trial particle will fit inside the methane hydrate cells – parts of a standard unit cell will be considered void. For high r_p, the crack is not wide enough for a trial particle to fit in. The length of the line segments is unimportant compared to the size of the trial particle, but consistent with intuition: The measured crack surface area slightly increases when Δl decreases.

6.4 Stress measurement

I output the stress tensor for each particle, usually averaged over some hundreds of integration steps. These values are put on a 2D grid, and averaged for each cell on that grid. This kind of measurement is not necessarily useful for methane hydrates, as the characteristic size is large (12 Å) so that the grid must necessarily be coarse, but the measurement can be interesting to illustrate the stress field, and check that it is qualitatively similar to the stress fields in continuum fracture mechanics. It turns out that the best results are obtained by using the same number of grid points as there are unit cells in each direction, which is not surprising, since This erases the inhomogeneities occurring within each unit cell. Slightly lower or higher resolutions of the grid result in serious artifacts and much higher resolutions results in very few atoms contributing to each grid cell, which gives bad statistics and useless results. The stress field from a representative simulation of a strained methane hydrate with a crack in it, measured with different resolutions of the averaging grid, is shown in figure 6.4.

6.5 Other tools

In addition to the tools already mentioned, I have implemented several shorter pieces of code to create plots and to calculate various properties like the diffusion constant and the viscosity in the model. I choose not to describe these in detail.

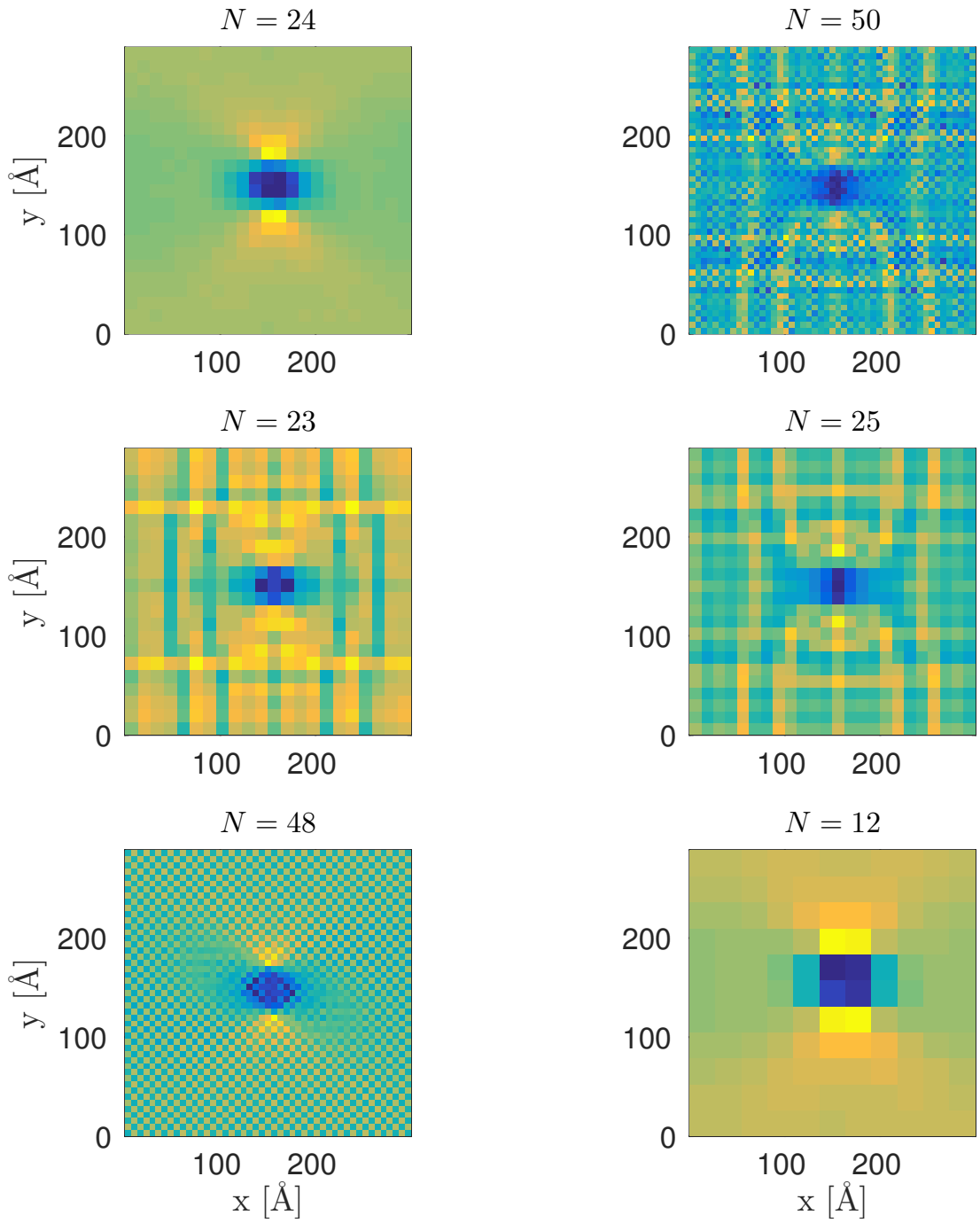


Figure 6.4: Measured stress fields in the xy-plane with different resolutions of the cell grid. The modeled system is $24 \times 24 \times 12$ sI unit cells. We see that $N = 24$ and $N = 12$ works well. $N = 48$ gives a checkered pattern, but it is possible to see the stress field. The highest resolution available without artifacts is the number of unit cells in each direction.

Chapter 7

Verification of models in LAMMPS

In this chapter, I verify that my LAMMPS setup reproduces known results from the literature. This is necessary to trust the model to go further with modifications.

I use potentials that are already parts of the LAMMPS distribution, so I only need to test that I use LAMMPS correctly.

7.1 TIP4P/Ice

I want to check that the TIP4P/Ice water potential in LAMMPS reproduces known thermodynamic properties from the literature. The TIP4P potential that comes with LAMMPS, really just handles the massless charged site; the rest of the implementation, namely the rigid bonds and angle, are implemented by the user. Additionally, parameters have to be set. If thermodynamic properties are reproduced, I can be confident that my configuration of the TIP4P potential in LAMMPS is really the potential introduced by [1].

7.1.1 Density of bulk water

As a quick check of my TIP4P/Ice setup, I want to check the density of bulk water in an NPT simulation. This property is easily measured, as long as the simulation time is sufficient to gather enough statistics.

After a few trial simulations, it turns out that the density of water is not a very good way to check whether the implementation is correct, since the values are very

Table 7.1: Liquid densities at melting points and melting points for several rigid water models at $P = 1\,\mathrm{bar}$. Adapted from [1]

Model	Melting point [K]	Density [$\mathrm{g\,cm^{-3}}$]
TIP4P/Ice	272.2	0.985
TIP4P	232.0	1.002
TIP4P/Ew	245.5	0.992
SPC/E	215.0	1.011
Expt.	273.15	0.999

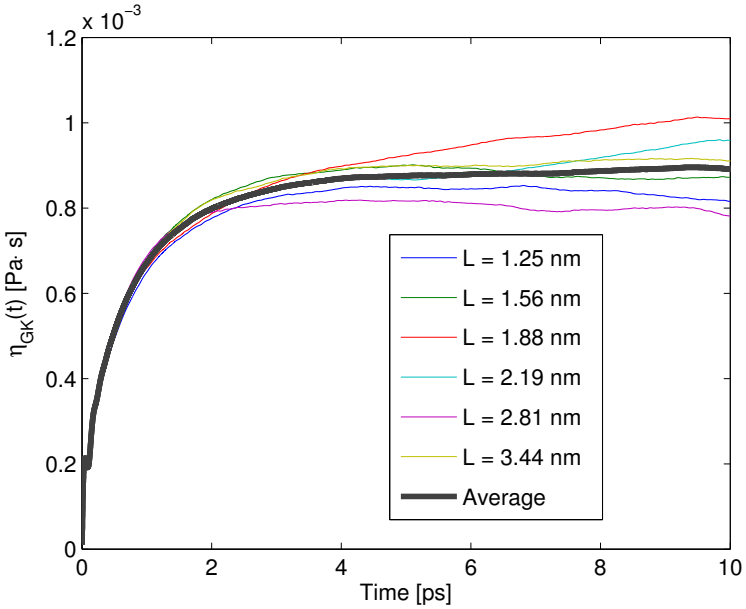


Figure 7.1: Shear viscosity for TIP4P/2005 at 300 K and $\rho = 0.98\,\mathrm{g\,cm^{-3}}$. The viscosity is estimated to $\eta_{GK} = 0.89(3)\,\mathrm{mPa\,s}$. Note the little bump all to the left in the plot, which corresponds to a little period of negative values of the autocorrelation function. Seeing this little bump verifies that the time resolution of the measurement is sufficiently fine.

similar in the different water models. I have a hard time estimating the density with sufficient confidence to separate it from the density of another water model subjected to the same conditions. Therefore, I go on to investigating more sensitive properties of the water model.

7.1.2 Diffusivity and shear viscosity

It is well established that the relative values of viscosity and diffusivity vary greatly between different water models, see eg. [27, 69]. That makes these quantities well suited for model verification. To find the shear viscosity η, I apply equation 4.32. Since i do not have access to infinite time series, i plot $\eta(t)$, and use the value it takes when it stabilizes as my estimate. To check that my LAMMPS implementation is correct, I use TIP4P/2005 parameters. Figure 7.1 shows the estimated shear viscosity from several NVT simulations of 2 ns with a density of $\rho = 0.98\,\mathrm{g\,cm^{-3}}$ and temperature $T = 300\,\mathrm{K}$. η is estimated at $t = 8\,\mathrm{ps}$ since the variation between the simulations is significantly larger than the fluctuations in the mean at this value of t. Reference values for shear stress and diffusivity are $\eta = 0.83(5)\,\mathrm{mPa\,s}$ and $D_0 = 2.49(6) \times 10^{-9}\,\mathrm{m^2\,s^{-1}}$ from [69].

In principle, the same simulation can be used to calculate the diffusivity, using the Einstein relation 4.31. But long simulations are not the way to go when estimating diffusion. An average over several shorter simulations, around 300 ps, gives better estimates. In Figure 7.2 I use a weighted linear regression on the diffusion coefficient for different system lengths. As mentioned in the theory section, the diffusion coefficient

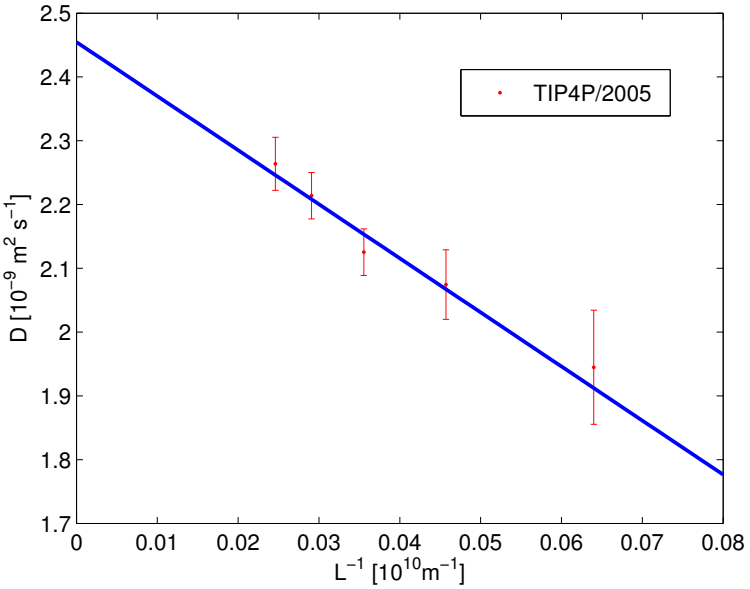


Figure 7.2: Diffusivity for TIP4P/2005 for simulations with the same conditions as in Figure 7.1. The diffusivity is estimated to $D_0 = 2.46(7) \times 10^{-9}\,\mathrm{m^2\,s^{-1}}$.

in a periodic molecular dynamics system depend on the box size, and this finite size effect on it goes like L^{-1}. Underlying each (red) data point in Figure 7.2 are $N_s(L) = 9$ simulations of 320 fs. Each of these simulations contribute an estimate of the diffusion coefficient using the Einstein relation on the mean squared displacement, and the mean of these $N_s(L)$ estimates is plotted. Error bars are estimated using the standard deviation of the $N_s(L)$ estimates:

$$e(L) = \sqrt{\frac{\mathrm{var}(D(L))}{N_s(L)}} \tag{7.1}$$

The expectation value of this error should go like $e \propto L^{-3/2}$ (inverse of the square root of the number of particles), which could have been another way to determine the weights. Then, the regression line is found with weighted linear regression where the weights are:

$$w(L) = \frac{1}{e(L)^2} \tag{7.2}$$

Using this procedure, I estimate the diffusivity in the TIP4P/2005 model at $T = 300\,\mathrm{K}$ and $\rho = 0.98\,\mathrm{g\,cm^{-3}}$ to $D_0 = 2.46(7) \times 10^{-9}\,\mathrm{m^2\,s^{-1}}$, which agrees well with the reference value. For completeness, Figure 7.3 contains the data underlying the diffusion coefficient calculations. It is included to show how noisy the data are – the model verification would not be satisfactory with just one simulation for each length of the simulation box.

Having measured both the shear viscosity and the diffusivity well within the uncertainty of reference values, I trust that my input files for the water model are correct, and go on to checking the methane model.

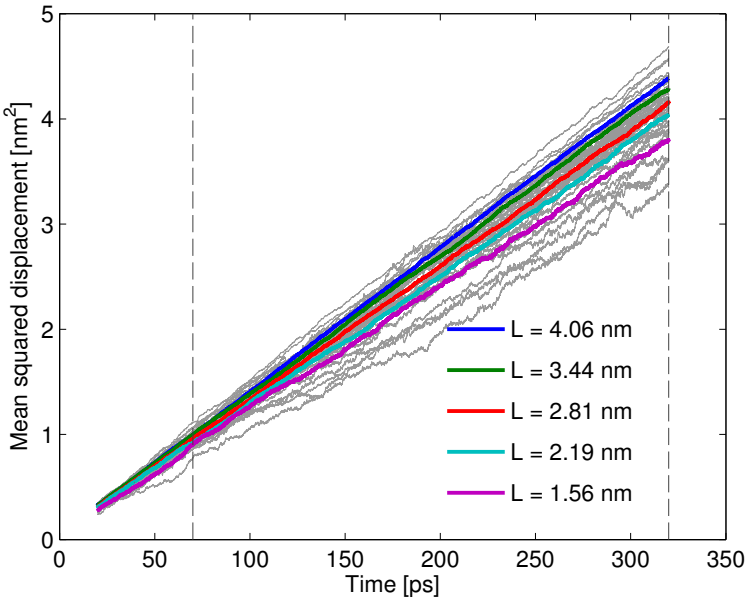


Figure 7.3: The data underlying Figure 7.2 (grey, thin lines) and the mean squared displacement averaged over N_s simulation for each system length L (colored, thick lines). Dashed vertical lines indicate the timeframe over which the diffusion coefficient was estimated.

7.2 United atom methane

The methane model is simply a Lennard-Jones model, and I will use it with a cutoff and no long-range corrections. Young's modulus and Poisson's ratio for the Lennard-Jones model will be verified when I check my protocol for measuring elastic properties in molecular dynamics. I am confident that LAMMPS treats the Lennard-Jones potential correctly, and I will have confirmation that my parameter input is correct if Young's modulus is correct (which it turns out to be).

7.3 Stabilizing methane hydrates

The TIP4P/Ice potential should be able to stabilize a methane hydrate structure. Therefore, I prepared an S1 hydrate using positions provided by Takeuchi *et al.* [68]. These positions were derived using the TIP4P (no suffix) potential, and are not expected to be an equilibrium configuration for TIP4P/Ice. The hydrate turns out to equilibrate nicely – although some water molecules turn around – using a Nosé-Hoover NPT thermostat with a temperature rising from 0 to 30 K. The reason for using a rising temperature rather than an energy minimization, is that the energy minimization schemes don't work with the SHAKE-algorithm in LAMMPS.

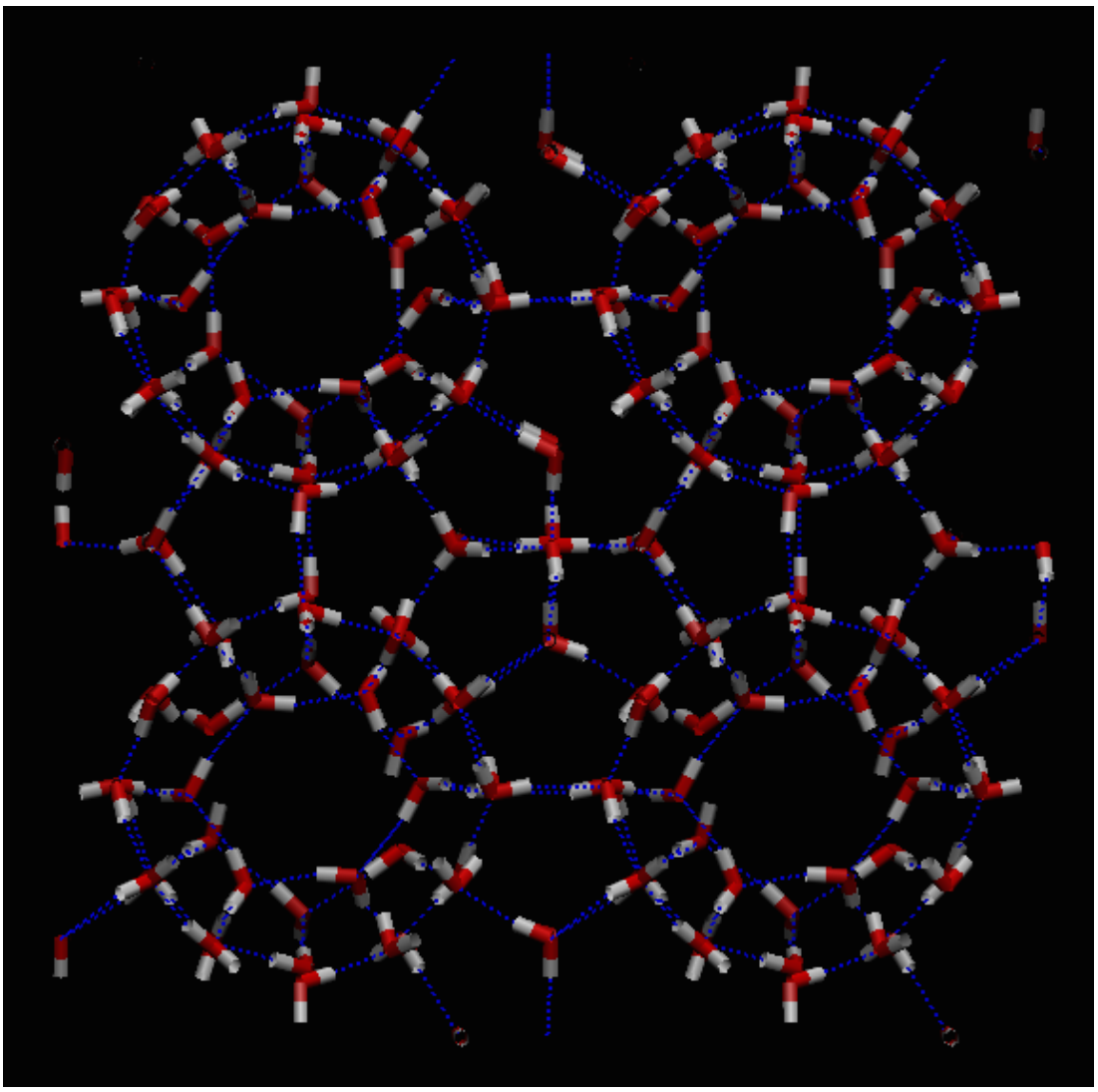


Figure 7.4: Methane hydrate being stable at low temperature.

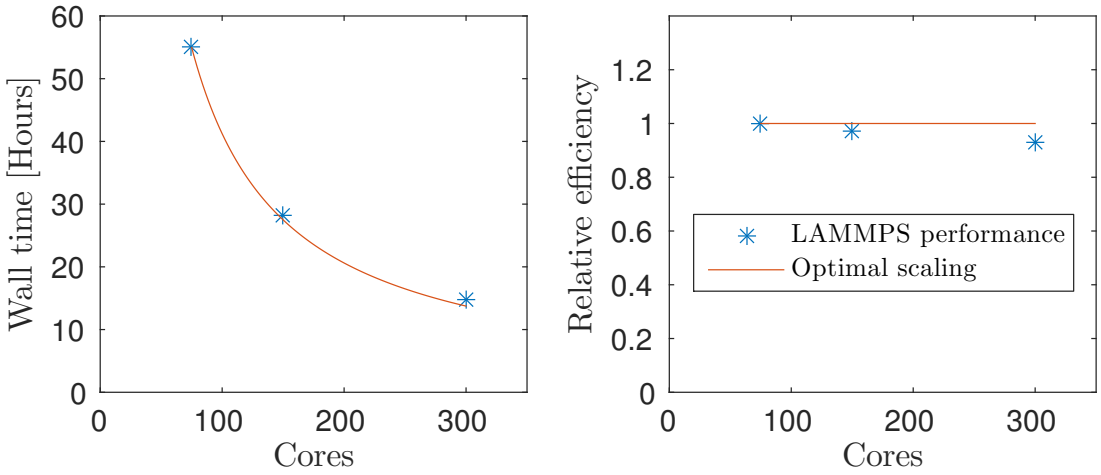


Figure 7.5: Wall time (left) and relative computational efficiency (right) for an sI methane hydrate system with approximately 10^6 atoms.

7.4 Numerical efficiency

Increasing the number of processor cores running a simulation almost always mean each CPU will work less effectively on the numerical task. But the wall time goes down, which means human waiting time goes down. The total price of a CPU-hour these days is around $ 0.10, which means they have to be carefully spent. Since human waiting time is not very critical in this project – the results won't be applied immediately like in weather forecasting – I need to find a reasonable balance between computational efficiency and human waiting time. Figure 7.5 shows the numerical efficiency of simulating a large system on up to 300 CPU cores.

7.5 Atoms out of range during P^{3}M calculation: A time-consuming bug

Since dealing with technical problems is a huge part of computational physics, I feel it necessary to describe some of the problems I have experienced. The most painful error that I experienced after getting rid of the beginners mistakes, was the following:

```
ERROR: Out of range atoms - cannot compute PPPM (../pppm_tip4p_omp.cpp:385)
```

This is painful to debug. It typically shows up in large simulations after a long time, and with no clear reason. Molecular dynamics simulations are expensive and time-consuming, and each attempt on a solution to this problem typically requires hours even on a supercomputer.

LAMMPS defaults to rebuilding neighbor lists every 10 timesteps. In some cases this is not sufficient when using TIP4P with PPPM, so the standard solution to this is to force more frequent neighbor list builds:

```
1 neigh_modify delay 0 every 1 check yes
```

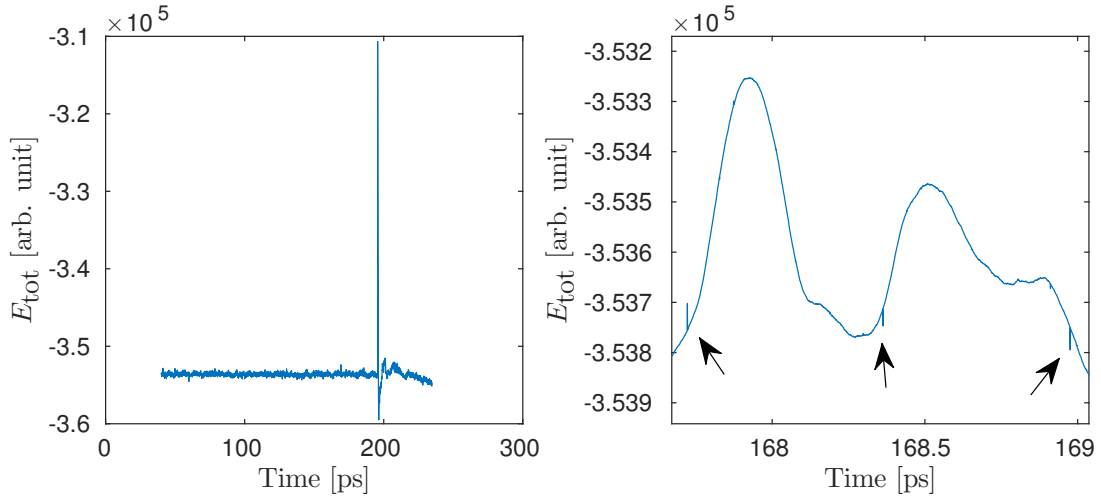


Figure 7.6: Little spikes (indicated by arrows) in the total energy curve hint that something is wrong with the simulation. These spikes disappeared when I stopped using the USER-OMP package. Note the small time-window in the right panel. These spikes are only visible when zooming closely in on the data.

However, this did not help me when I got the message. The next thing I tried was to divide the timestep by two, which did not help either.

The next approach is to try to increase the damping time of the thermostat, as thermostats with a low characteristic time are known to cause trouble. But this did not help either. Then I realized that I did not experience this problem when I did not change from NPT to NVT during the simulation. Something ugly may happen when this is done.

To avoid doing NPT and NVT in the same simulation, I can start by creating an equilibrated system containing a crack, and use the simulation box size and particle positions from that simulation as input in my crack simulations. This of course also introduces the advantage of not having to equilibrate the system for every simulation. However, it also introduces the disadvantage of needing a separate file for each initial crack size.

It turned out that the problem was related to the OpenMP package bundled with LAMMPS. Some bug is introduced when using NPT and the OpenMP package. The OpenMP package did not give vital efficiency improvements, so I choose not to use the OpenMP package, rather than resolving the error.

There are signatures of this error in the total energy curve of the system. Figure 7.6 shows the total energy of an sI system while waiting for a crack to propagate. The simulation failed with the error message described above. As indicated in the figure, small discontinuities can be spotted in the total energy curve. For this particular simulation, there was also a large spike close to the end of the simulation, but that kind of spike doesn't show up often.

Chapter 8

Modeled systems

In this chapter, I work out how to do simulations of the system described in chapter 5, and analyze simulations in the pursuit of answering the questions raised in chapter 2.8.

The first step is to work out basic properties of the model I study. Curiously, I have not found any records of the shear viscosity and the self-diffusion coefficient in the TIP4P/Ice water model, so I start by calculating those quantities. Since studies of mechanical properties of methane hydrates modeled with TIP4P/Ice + united atom methane (UAM) are scarce, I continue by working out even the most basic mechanical properties of the model: Poisson's ratio and Young's modulus. In order to be sure that the methods I choose for estimating the mechanical properties work, I start by checking the methods on a Lennard–Jones crystal, and benchmark my results against known parameters from the literature. Then I apply the methods to calculate the mechanical properties of the methane hydrate. When the mechanical properties are obtained, I continue with studying fracture in methane hydrates.

8.1 Shear viscosity and diffusivity of the water model

I have not succeeded in finding values for the shear viscosity and diffusivity of bulk liquid water modeled with TIP4P/Ice. To measure these properties, I run simulations like the ones I ran to calculate and verify the same properties for the TIP4P/2005 potential, namely bulk simulations of water. Figure 8.1 shows the Green-Kubo relation for the shear viscosity (equation 4.32) using 5 independent pressure components from 4 independent simulations with different simulation box sizes. The 5 pressure components are averaged and go into a single line, so each line in the figure represents the average of the pressure components of a single simulation. Based on these data, I estimate the shear viscosity of TIP4P/Ice to $\eta_{GK} = 1.63(5)\,\mathrm{mPa\,s}$, where the uncertainty is estimated in the same way as for the equivalent calculation on the TIP4P/2005 model; as the standard deviation arising from the set of values obtained by reading the value of the integrated autocorrelation function at 8 ps for each simulation. The self-diffusion constant is calculated with the Einstein relation (equation 4.31): Figure 8.2 shows the self-diffusivity as a function of the inverse simulation box length in the TIP4P/Ice-model on the same axes as the results for TIP4P/2005 that was obtained in chapter 7. The self-diffusion constant of TIP4P/Ice is estimated to $D_0 = 4.0(1)\,\mathrm{m^2\,s^{-1}}$.

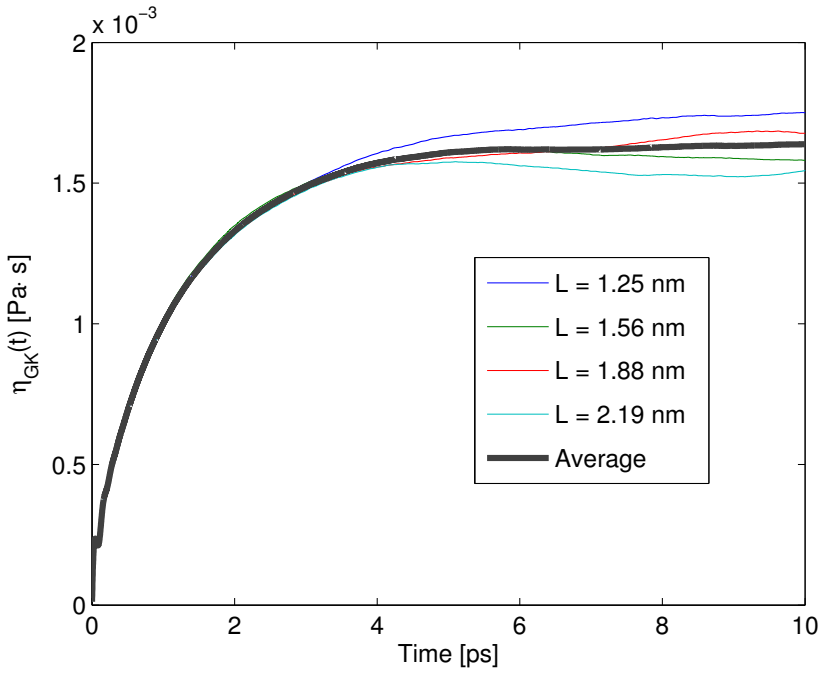


Figure 8.1: Shear viscosity for TIP4P/Ice at 300 K and $\rho = 0.98\,\mathrm{g\,cm^{-3}}$. Each thin line shows the average of the 5 independent pressure component. The thich line is the average of 4 independent simulations. The viscosity is estimated to $\eta_{GK} = 1.63(5)\,\mathrm{mPa\,s}$

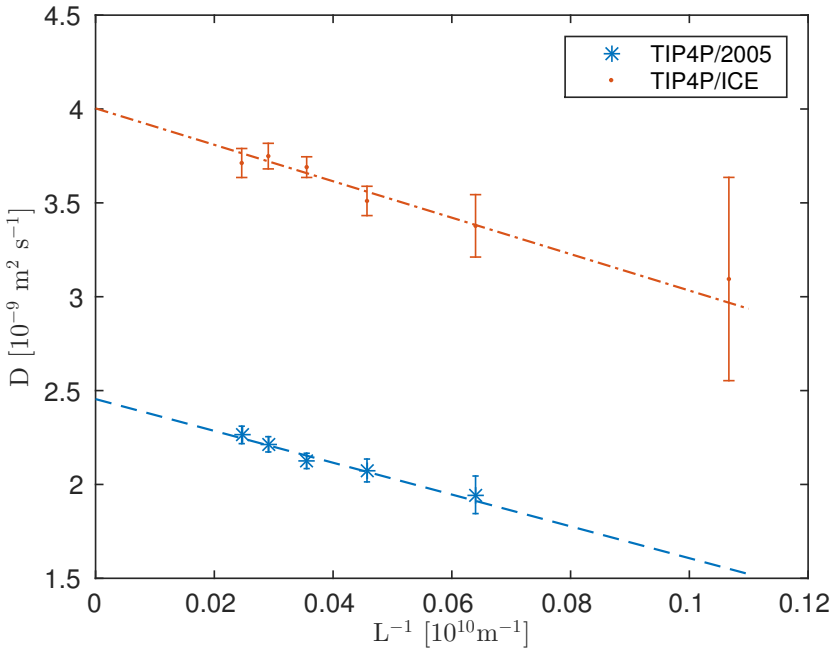


Figure 8.2: Linear regression of self diffusion coefficients as a function of inverse simulation box length. Results for TIP4P/2005 (blue) were already reported in the verification section. The self-diffusion coefficient for TIP4P/Ice is estimated to $D_0 = 4.0(1) \times 10^{-9}\,\mathrm{m^2\,s^{-1}}$

8.2 Measuring elastic properties with a constant strain rate

A simple approach to measuring elastic properties, which I will use for my estimates, is to subject a system to a constant strain rate by expanding the simulation box in one of the coordinate directions. Atom positions must be rescaled accordingly, to avoid the introduction of elastic waves. This is also justified by the fact that there are no "special places" in the simulation box – since it is periodic in all directions – so expanding the simulation box without rescaling the particle positions would be a strange choice. An anisotropic barostat will be applied along the other axes, to keep the confining pressure constant. The barostat will change dimensions of the simulation box perpendicular to the applied deformation, so the lateral strains can be measured using the dimensions of the simulation box:

$$\varepsilon_i = \frac{L_i - L_{0,i}}{L_{0,i}} \tag{8.1}$$

If the sample is isotropic, strain application in only one of the coordinate directions has to be applied to estimate both Young's modulus and Poisson's ratio for a given system. To account for the possibility that the model I use do not exhibit isotropic behavior, I will deform the system in only *one* direction, but check the resulting lateral strains in both of the coordinate directions perpendicular to the axis of applied strain. The value of Poisson's ratio,ν, is minus the ratio of lateral to normal strain. For linear materials, this can be calculated as the negative of the slope of the strain-strain curve.

The value of Young's modulus,E, is normal stress divided by normal strain. Again, this can be calculated as the slope of the stress–strain curve if the material is linear. I will calculate these slopes using linear regression with least squares on the curves, and check whether the slope is actually linear by visual inspection.

In the following, I apply the method described above to a Lennard-Jones crystal and to the TIP4P/Ice+UAM methane hydrate model.

8.2.1 Lennard-Jones crystal

The FCC-lattice with a Lennard-Jones potential has been extensively investigated due to its simplicity and its fundamental role in molecular dynamics. Therefore, it provides robust benchmarking capabilities. I will check that my protocols for dynamic (but quasi-static) determination of elastic properties reproduce known parameters for a Lennard-Jones solid. Reference values are: Young's modulus, $E = 61.1\epsilon/\sigma^3$ ($= 2.40\,\mathrm{GPa}$ for the parameters I use for methane), and Poisson's ratio, $\nu = 0.347$. These values are taken from a molecular dynamics study by Quesnel *et al.* [59].

The elastic test is performed on two Lennard–Jones systems of different size: One consisting of 11^3 FCC unit cells, and another consisting of 22^3 FCC unit cells. The samples are subjected to a strain rate of $2 \times 10^{-8}\,\mathrm{fs}^{-1}$ over 0.4 ns, resulting in a maximum strain of 8×10^{-3}. The external pressure is set to 50 MPa, and the temperature is 5 K. The stress–strain curve and the normal strain–lateral strain curves are shown in figures 8.3 and 8.4, respectively. Estimates of Poisson's ratio and Young's modulus, taken as the best linear regression with least squares, are indicated in the figure legends. The data show no significant finite-size effects on the elastic properties. Young's modulus is estimated to $E = 2.48\,\mathrm{GPa}$, and Poisson's ratio is estimated to $\nu = 0.35$. This corresponds well with the numbers reported by Quesnel *et al.* [59].

8.2.2 sI methane hydrate with TIP4P/ICE+UAM

To my knowledge, there are no published estimates of Young's modulus and Poisson's ratio for the TIP4P/Ice+UAM model of methane hydrates. Therefore, I seek to make crude estimates of these quantities in dynamic simulations. I perform the elastic test outlined above, with zero confining pressure. The test is performed on two systems of 11^3 sI unit cells. One subjected to a strain rate of $5 \times 10^{-7}\,\mathrm{fs}^{-1}$, and another subjected to a strain rate of $2 \times 10^{-7}\,\mathrm{fs}^{-1}$. Having performed two simulations with a different strain rate, the calculated mechanical properties can be extrapolated from these simulations to infinitely slow strain, yielding more accurate estimates. Figure 8.5 shows the stress–strain relationships and corresponding estimates of Young's modulus. By extrapolation, Young's modulus is estimated to $E = 7.1\,\mathrm{GPa}$. Figure 8.6 shows the relationship between applied normal strain and the measured laterals strains. Based on the results shown in this figure, I find it reasonable to treat methane hydrates as isotropic for this work. My extrapolated estimate of Poisson's ratio is $\nu = 0.41$. Putting the calculated values for Young's modulus and Poisson's ratio, along with the density calculated from Ning *et al.* [52], $\rho = 0.919\,\mathrm{kg\,L}^{-1}$, into the equations for shear and pressure waves, give the elastic wave speeds for the methane hydrate: $v_s = 3780\,\mathrm{m\,s}^{-1}$ and $v_p = 1650\,\mathrm{m\,s}^{-1}$. From the Poisson's ratio and the shear wave speed, the Rayleigh

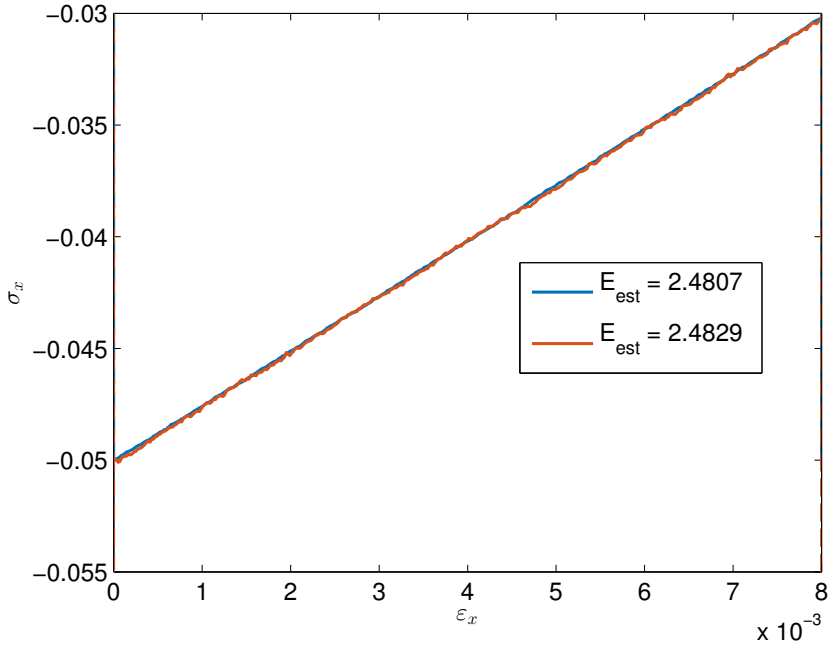


Figure 8.3: Stress-strain relations for Lennard-Jones systems of 11^3 (red) and 22^3 (blue) FCC unit cells. The sample was subjected to a constant strain rate of $2 \times 10^{-8}\,\mathrm{fs}^{-1}$. Young's modulus, E, is estimated using linear regression with least squares on all data points.

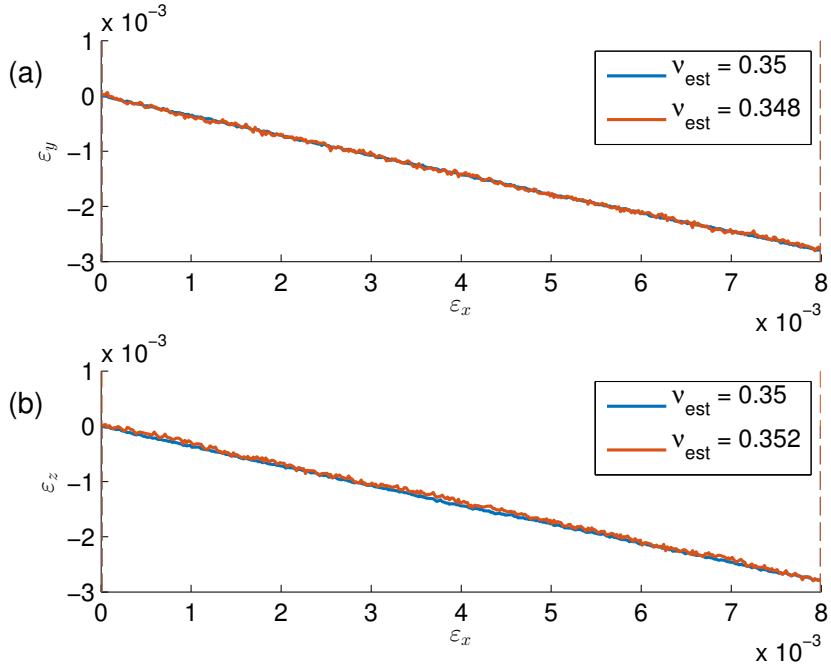


Figure 8.4: Strain-strain relations for for the same simulations as in Figure 8.3. All data points were used to estimate Poisson's ration,ν.

Table 8.1: Mechanical properties of the methane hydrate in the TIP4P/Ice + UAM model measured under isothermal tensile strain with zero confining pressure. The elastic wave speeds are derived from Young's modulus and Poisson's ratio.

Property	Value
Young's modulus, E	7.1 GPa
Poisson's ratio, ν	0.41
Shear wave speed, v_s	$3780\,\mathrm{m\,s^{-1}}$
Pressure wave speed v_p	$1650\,\mathrm{m\,s^{-1}}$
Rayleigh wave speed v_R	$1570\,\mathrm{m\,s^{-1}}$

wave speed can be calculated using the following approximate formula from [46]:

$$v_R = v_s(0.874 + 0.196\nu - 0.043\nu^2 - 0.055\nu^3), \tag{8.2}$$

This results in a Rayleigh wave speed of $v_R = 1570\,\mathrm{m\,s^{-1}}$. The new mechanical properties obtained for methane hydrates in the TIP4P/Ice + UAM model in this work are given in table 8.1. Compared to the experimental values I presented in table 2.1, these mechanical properties are quite good, except that Poisson's ratio differs quite significantly from the experimental value.

8.3 Early results on fracture and fracture toughness

Having obtained the basic elastic properties, I go on to the fracture studies. I start by discussing the simulation protocol for fracture simulations, and continue by performing preliminary simulations to verify that the protocol works. The main goal I work towards here, is to calculate the fracture toughness of methane hydrates.

8.3.1 Simulation protocol

Several approaches can be applied to determine the fracture toughness in molecular dynamics. Hantal *et al.* [30] performed NVT simulations and imposed incremental deformations to their sample. After each deformation, they minimized the system, and then ran molecular dynamics for 10 ps. I tried to use the same protocol, but found that for my system, the impact on the energy distribution among the degrees of freedom was too large when performing minimizations between the deformations. After a deformation, the system is no longer in equilibrium, and the minimization is supposed to bring the system closer to equilibrium. However, in an equilibrium of finite temperature, some fraction of the system energy is associated with moving particles being temporarily positioned closer or farther from each other than the equilibrium distance dictated by their inter-atomic potential and the overall structure of the system. This is a thermal energy. A minimization removes this energy from the system, and when an NVT simulation is started after minimization, the thermostat will have to put this energy back into the system. This takes time: With the thermostat settings I apply, it takes several tens of picoseconds. Therefore, a possible protocol that does not use minimization is:

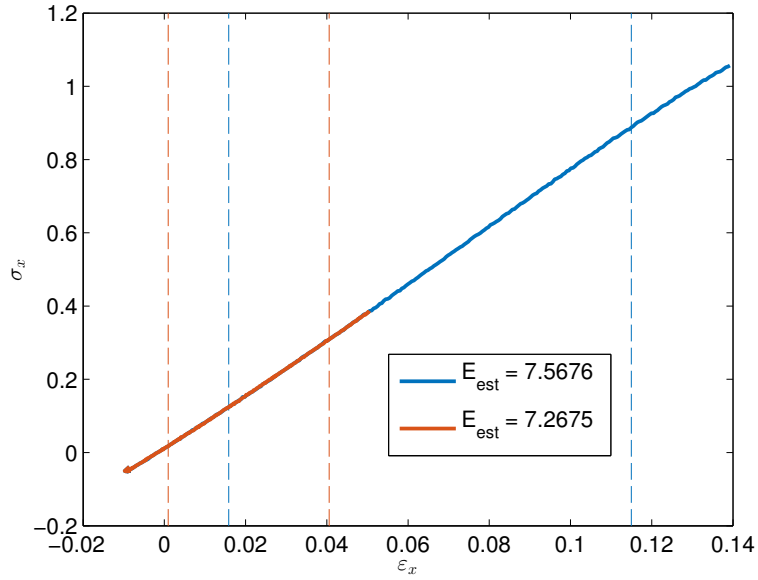


Figure 8.5: Stess–strain relations for a system of 11x11x11 sI unit cells. Dashed lines indicate the region that was used to estimate Young's modulus. Strain rates of $5 \times 10^{-7}\,\mathrm{fs}^{-1}$ (blue) and $2 \times 10^{-7}\,\mathrm{fs}^{-1}$ (red) along the x-axis. Upon close visual inspection, a slight rising slope can be seen for small strains and a rising slope for large strains, but overall the stress-strain relation is surprisingly linear, especially given the large strain range.

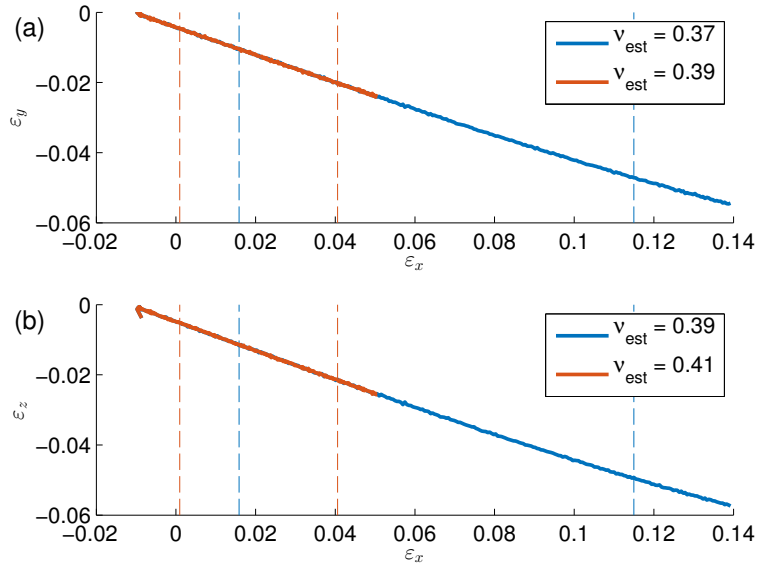


Figure 8.6: Strain-strain relations for the same system as in Figure 8.5. Measured strain along the y-axis (a) and z-axis (b) is plotted against the applied strain along the x-axis. Dashed lines indicate the region that was used to estimate Poisson's ratio.

1. Carve out an initial crack and equilibrate the system NPT to relax the system after the crack is introduced.
2. Perform small deformation
3. Re-equilibrate the system NVT.
4. Molecular dynamics run (NVT) to wait for failure
5. Return to 2 or end the simulation.

This approach was also tested, but it turned out that the waiting times were unpredictable – it was hard to set a reasonable waiting time for step 4. This resulted in additional deformations being applied *during* fracture.

The protocol I will actually use: A simpler protocol – and I believe this to be a good way to study this particular system – is to subject the sample to a constant strain rate until it reaches some predefined strain, and then wait for a crack to propagate. This is simple to do, and it is also close to experimental conditions, except that the strain rate must necessarily be much higher in MD simulations than in experiments. The downside of this method is that only a single strain level will be tested in each simulation, but if there exists waiting times for fracture that depend on the applied strain, then there has to be performed individual simulation for each strain level anyway.

In addition to complication the simulations, I see the possible waiting times for fracture as a research opportunity. Waiting times for fracture can depend for example on the applied strain and temperature, and this relation can be characterized. It will probably be very computationally demanding to obtain good statistics, but I believe that some tendencies can be identified without spending too many CPU-hours.

8.3.2 Proof-of-concept simulations

Before performing large scale simulations, I do preliminary, smaller, simulations to find out whether the protocol proposed in the latter section can produce useful results. I do simulations on the thinnest possible system consisting of sI unit cells – the system that is only one unit cell thick – to keep computational costs down. The total computational cost for tuning in on parameters and getting rid of bugs and blunders has been around 10^4 CPU hours.

The proof-of-concept simulations are performed using the protocol outlined above (the one that I state I will be using), on systems of $24 \times 24 \times 1$ sI methane hydrate unit cells. The systems are subjected to a constant strain rate during 50 ps, to obtain strains ranging from 0.045 to 0.1. Then, the system is left on its own. Table 8.2 shows all parameters of these simulations.

The simplest way to identify whether a crack has propagated, it turns out, is to check the potential energy curve. The potential energy of the system goes down when the system fractures. Figure 8.8 shows the potential energy of the proof-of-concept

Table 8.2: Parameters for the proof-of-concept simulations.

System	$24 \times 24 \times 1$ sI unit cells. Initialized with crack of 6×40 Å
Protocol:	0–100 ps: NPT @ 101 325 Pa, 260 K
	100–150 ps: NVT @ 260 K with straining
	150–550 ps: NVT @ 260 K and final strain
Final strain:	[1.045, 1.05, 1.055, 1.1]
Interactions:	TIP4P/Ice and United Atom Methane
Short-range potentials:	Lennard–Jones and Coulomb with 10 Å cutoff.
Long-range corrections:	Coulomb with P^3M allowing a relative error of 10^{-4}.
Thermostat damping time:	1 ps
Barostat damping time:	1 ps
Integration timestep:	1 fs
SHAKE tolerance:	10^{-4}
Drag term (LAMMPS specific):	1.0

simulations. In three of the systems, a crack spans the yz-plane after the simulation is finished. In one of the simulations, no crack growth is observed.

Below is a description of the visual experience of the crack propagation in one of the simulations:

Visual experience of the crack propagation (figure 8.7) First, the system slightly contracts (NPT allows volume change). The system comes to rest, and some methane molecules diffuse out of near-wall cages to the hole that was carved out during initialization. After 100 ps, straining starts. The system expands to 1.05 times its original length along the x-axis at a constant rate during 50 ps. This corresponds to a very high strain rate in macroscopic terms. More methane fills the crack, as its volume increases when the crack gets wider – but the crack length remains the same during expansion. The system has now reached the desired strain. Methane molecules bounce back and fourth inside the crack, and the crack edges seem jittery. Suddenly, a hydrogen bond near a crack tip breaks – fluctuations from methane molecules and lattice vibrations have made the system unstable. When the first bond breaks, the next bond along the crack axis cannot sustain the stress. Bonds near the other crack edge break. Then bonds break one after the other on both edges of the crack – the crack propagates. In a matter of tens of picoseconds, both crack edges reach the periodic boundary. During crack propagation, clathrate cages are ripped apart. Water molecules remain stuck to the wall, while methane is released, and fills the void between the two pieces of methane hydrate (To be more precise, there is still only one piece of methane hydrate because of the periodic boundaries). The crack is not traveling straight in one direction: It first starts traveling between columns of sI cells before it turns and continues to propagate in the middle of hydrate cells, ripping apart big cages. Just after the crack has propagated, the methane hydrate oscillates with a spatial amplitude comparable

to the void width. During the following hundreds of picoseconds, the oscillations damp out – the system is approaching a new equilibrium.

From the visuals, I move on to preliminary calculations of the surface energy and the critical energy release rate. The surface energy can be estimated as the change in potential energy from the equilibrated system before straining to the system of methane hydrate and free methane after crack propagation. The energy release rate can be estimated as the change in potential energy during crack propagation. If ΔU_c is the change in potential energy during crack propagation, and ΔU_s is the change in potential energy from before straining to after crack propagation, then the formulas for G_c and γ_s read:

$$\mathcal{G}_c \approx \frac{\Delta U_c}{L_y L_z} \tag{8.3}$$

$$\gamma_s \approx \frac{\Delta U_s}{2 L_y L_z} \tag{8.4}$$

Where the $\frac{1}{2}$ factor is because the crack opens two surfaces. The thermalization for all four of the simulations are equal, with an average potential energy of $-2.8112\,\mathrm{fJ}$ on the plateau (40–100 ps). The average potential energy after crack propagation is $-2.7970\,\mathrm{fJ}$, with almost no variation between the simulations. This yields an energy difference of $1.42 \times 10^{-17}\,\mathrm{J}$. Using the simulation box cross-sectional area (measured values: $L_y = 288.8\,\text{Å}$, $L_z = 12.04\,\text{Å}$) as an estimate of the crack size, the estimated surface energy is $\gamma_s = 0.204\,\mathrm{J\,m^{-2}}$. Since the energy level required to start a crack is not well defined from figure 8.8, the estimate of the critical energy release rate is coarser than that of the surface energy. From the figure, I read off a potential energy of around $-2.765\,\mathrm{fJ}$ from the simulation that needed the least strain to produce a system-spanning crack. The estimated critical energy release rate from this potential energy is $1.3\,\mathrm{J\,m^{-2}}$. This corresponds to, using equation 3.38 for the stress intensity in isotropic materials, a critical stress intensity factor of $K_{Ic} = 0.11\ \mathrm{MPam}^{\frac{1}{2}}$ (Using $E = 7.1\,\mathrm{GPa}$ and $\nu = 0.41$). Unfortunately, we shall see that this method for finding the fracture toughness is wrong, because of how the energy was measured. This will be discussed and corrected later in this chapter. For reference, the experimental fracture toughness of freshwater ice is around $K_{Ic} = 0.10\ \mathrm{MPam}^{\frac{1}{2}}$ [7].

Detailed crack analysis The estimate of the surface area of the crack based on the simulation cross-sectional area, is a rough estimate. I therefore decided to develop an improved measure of the crack area using a tailored algorithm and my own code. This algorithm was described in chapter 6. Hopefully, that estimate will be better than the estimate solely based on the simulation box dimensions. The code also lets me follow the crack area in time, which makes it possible to measure the crack speed. I choose to define the crack speed as the change of crack area divided by the crack depth, i.e. the length of the simulation box in the z-direction, L_z. Keeping in mind that my cracks travel in two directions, and that the crack opens two surfaces, the crack tip velocity

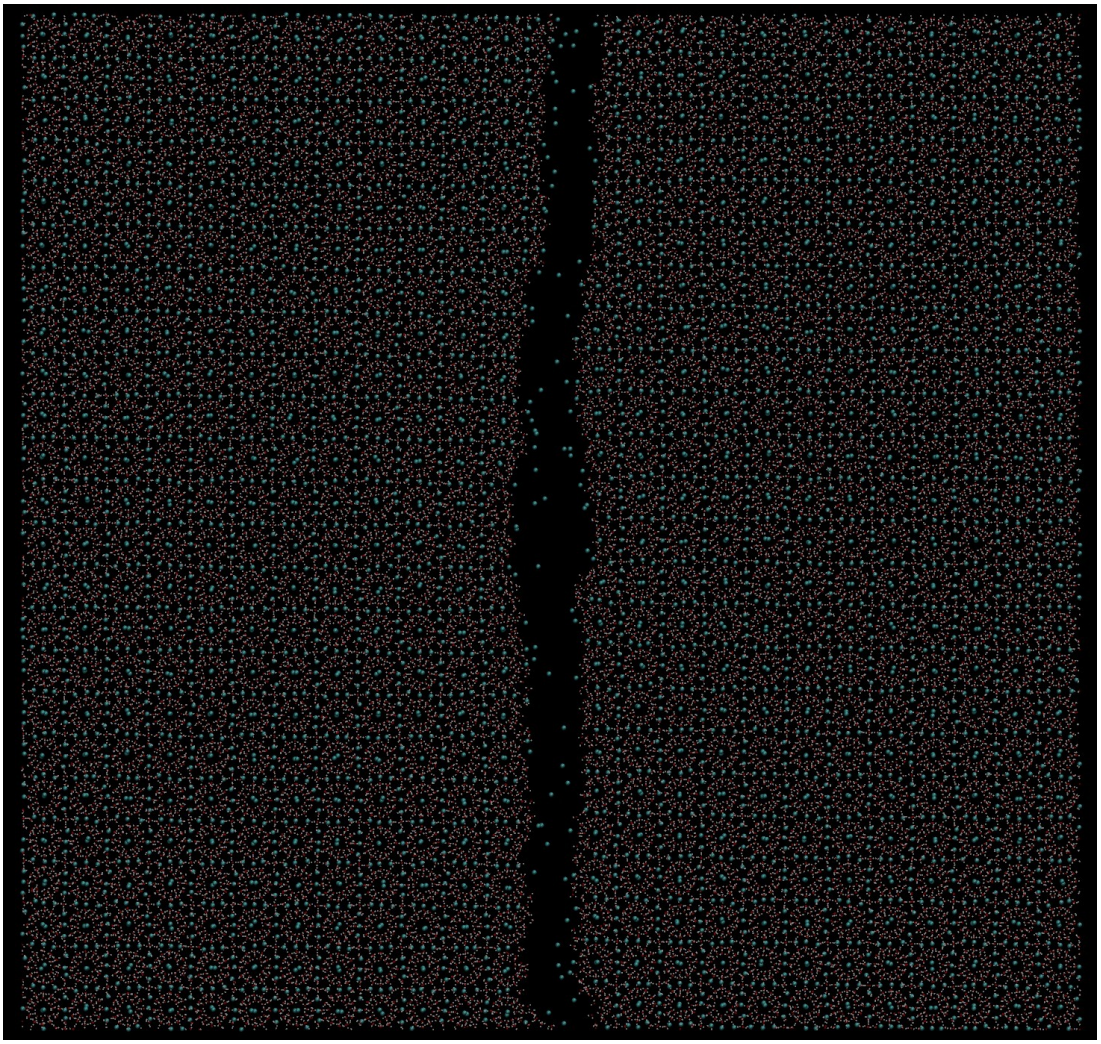


Figure 8.7: Fracture of a system of $24 \times 24 \times 1$ sI unit cells. The upper crack changes from propagating between columns of unit cells to propagating in the middle of a column of unit cells, breaking big cages instead of small ones.

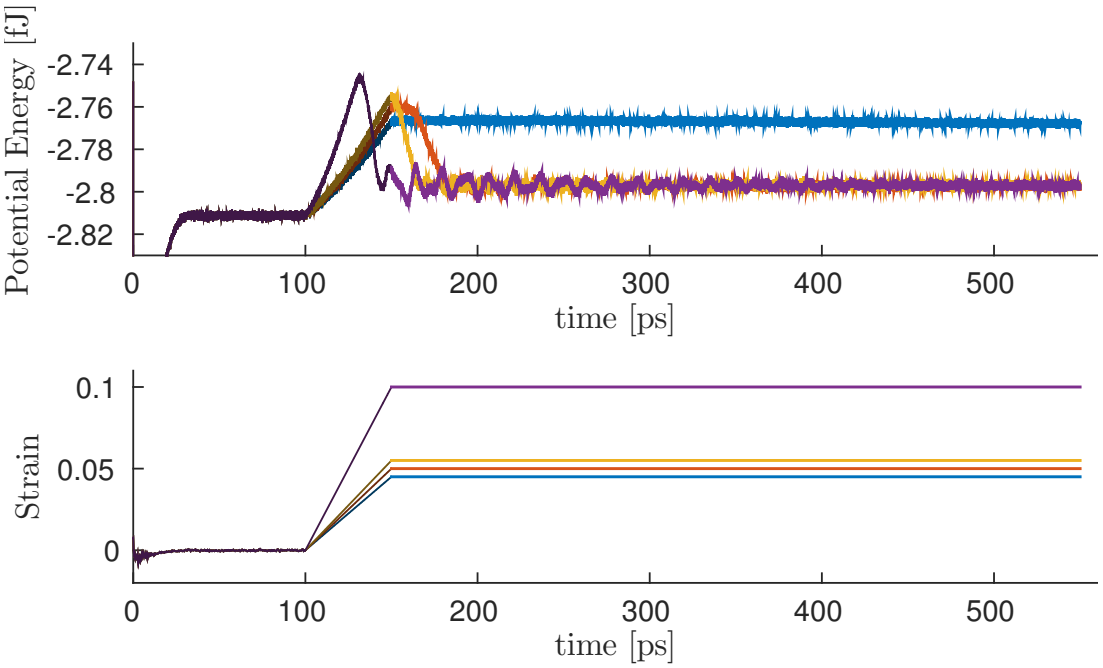


Figure 8.8: Potential energy (upper panel) and strain (lower panel) for a series of simulations where systems of $24 \times 24 \times 1$ unit cells of sI methane hydrate were subjected to tensile strain. The systems were initiated with an elliptic crack of 6.0×40.0Å. The system was then allowed to equilibrate NPT during 100 ps, which can be seen from the small fluctuations in strain in this time-frame. Then the simulations were continued NVT, but with an imposed volume change due to the application of a constant strain rate in the x-direction, taking the system to a predesignated stress after 50 ps of straining. Then, a regular NVT-simulation was run for 400 ps (bright colors). The thermostat and barostat damping times were 1 ps. $T = 260$ K. Rapidly falling potential energies are due to crack opening. The slope of falling potential energy is systematically steeper for higher values of the strain before crack propagation. The systems that fracture show an oscillating potential energy after crack propagation, which is due to global oscillations of the system. The oscillations are damped out by a drag term in the thermostat. (The parameters of this simulation are also given in table 8.2.)

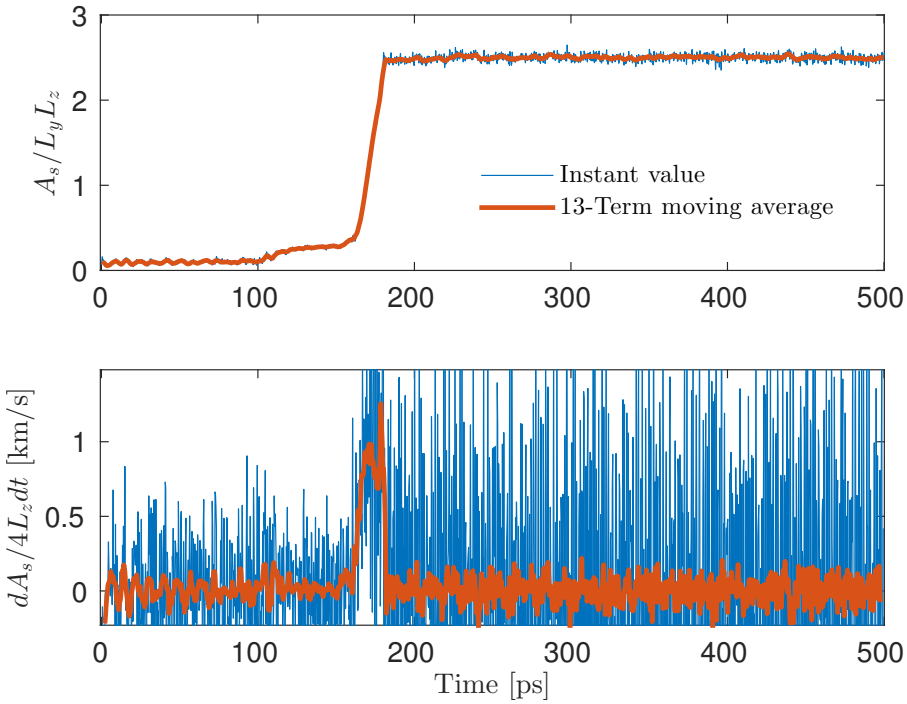


Figure 8.9: Evolution of crack area in time (upper panel) and derived crack tip speed (lower panel). The area is scaled by dividing by the cross-sectional area of the simulation box to be able to compare simulations of different thicknesses.

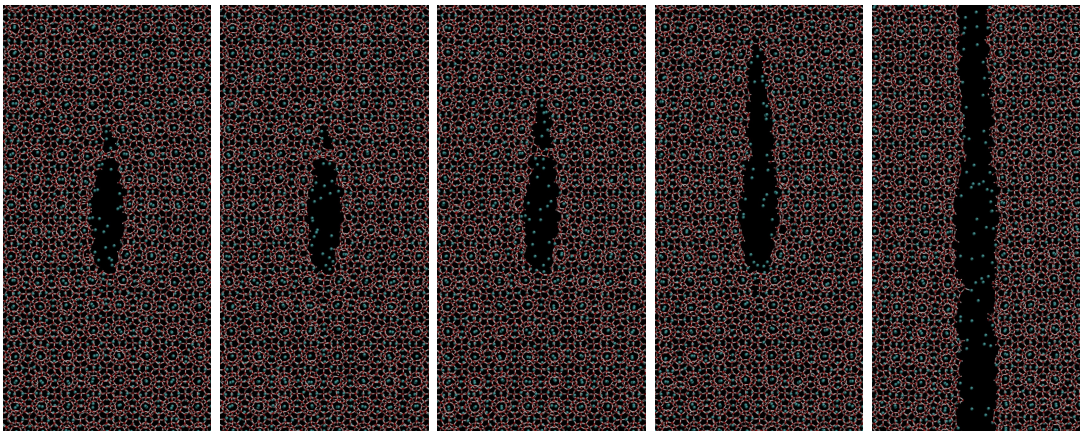
of the crack that was initially cut as an elliptical prism with its axis in the z-direction becomes:

$$v_c = \frac{1}{2L_z}\frac{\mathrm{d}A_s}{\mathrm{d}t} \tag{8.5}$$

Where A_s is the measured crack surface area. Figure 8.9 shows the evolution of the crack area and crack speed in time in one of the proof-of-concept simulations, using a standard 5-point stencil to calculate the derivative of the crack surface area.

Summary of proof-of-concept simulations The results from these initial simulations indicate that methane hydrates are very brittle on the tens-of-picoseconds scale, in the sense that they do not at all deform plastically to withstand strain. They either deform elastically, or they fail. There are also indications that the time from straining ends until fracture begins depends on the applied strain in a systematic way. Also, a close look at the potential energy curve of the simulation in figure 8.8 that did not end in rupture reveals that the potential energy is actually slowly decreasing. Whether this is a sign of a coming fracture, strengthening rearrangements of particles – which could imply some ductility – or something else, remains to be investigated.

Observation of special feature – a cavity in front of the crack tip Later on, I performed a few simulations with the same setup as in the proof-of-concept simulations,



(a) $t = 300\,\text{ps}$ **(b)** $t = 540\,\text{ps}$ **(c)** $t = 555\,\text{ps}$ **(d)** $t = 561\,\text{ps}$ **(e)** $t = 582\,\text{ps}$

Figure 8.10: Development of a cavity in front of the crack and then propagation of the crack.

but with other levels of the final strain. In one of these simulations, one with a final strain of 0.046, I discovered a strange feature: A cavity formed in front of the initial crack. The crack did not go on to develop rapidly within the total simulation time of 550 ps, so I continued the simulation to see if it would develop further with time. It turned out that this simulation only needed a few tens of picoseconds to develop a system-spanning crack. Pictures of the time evolution of this crack are shown in figure 8.10. Cavities in front of the major crack have been discussed by for example Bouchbinder *et al.* [12].

8.3.3 Arising questions

Several questions arise from the results of the proof-of-concept runs:

- Is the amount of methane that is freed during fracture propagation always the same?
- Is there a relationship between the waiting time from straining ends until rupture starts and the slope of the potential energy curve?
- Is the slowly decreasing potential energy of long waiting-time events important, and can it be related to the waiting time?
- What is the fracture toughness for long (infinite) waiting times, and is it possible to measure? Will the system melt before it fractures?
- Is the methane in the initial crack important for fracture initiation?

There are also several technical problems that can be addressed:

- What is the effect of the thermostat damping time on fracture properties?

- Is the strain rate important? Does the time it takes to strain the system compete with the waiting time until fracture?

Each of these questions will require substantial efforts to be answered, and I will only have time to address a few of them. In the sections to come, I focus on the crack initiation, and the amount of free methane before, during and after crack propagation. I will essentially ignore the technical issues for now. If I find something interesting, the technical issues will of course be part of the verification, but I cannot justify the computational cost of such verifications before I have found anything interesting, surprising or counterintuitive that needs thorough verification.

8.4 Energy considerations – the thermodynamics of expanding the simulation box

The result that $\mathcal{G}_c > 2\gamma_s$, as was found from the preliminary results was peculiar – and wrong. In this section, I make an argument for why the potential energy cannot be directly used to calculate the fracture toughness. I am probably not the first person to make this argument, but I have not seen it made explicitly. I have found papers, e.g. Hantal *et al.* [30], where the same kind of argument must have been made, either explicitly or implicitly, during the research process.

In the last section, I only considered the potential energy to estimate the fracture toughness, since fracture mechanics dictates that the fracture toughness should be the amount of potential energy released per projected crack surface area opened. But really, it is the Helmholtz free energy release rate that should be used for fracture toughness calculations – potential energy in fracture mechanics is not necessarily the same as in molecular dynamics. A more thorough analysis solves the problem of methane hydrates seeming very brittle while having $\mathcal{G}_c > 2\gamma_s$, and the conclusion, which we shall see, is that we actually have $\mathcal{G}_c \approx 2\gamma_s$.

Let us consider expanding the simulation box along the x-axis. This corresponds to performing work on the system:

$$W = \int_{t(L_x=L_x^0)}^{t(L_x^0+\Delta x)} \int_{yz} \sigma_{xx}(t,y,z)\ \mathrm{d}z\mathrm{d}y\mathrm{d}x \tag{8.6}$$

If the system is in equilibrium, then:

$$\int_{yz} \sigma_{xx}(t,y,z)\ \mathrm{d}z\mathrm{d}y = L_y L_z \Sigma_{xx}(t) \tag{8.7}$$

Where Σ_{xx} is an element of $\underline{\underline{\Sigma}}$, which is the mean stress tensor for particles in the system. The work on the system can then be written:

$$W = L_y L_z \int_{t(L_x=L_x^0)}^{t(L_x^0+\Delta x)} \Sigma_{xx}(t)\ \mathrm{d}x \tag{8.8}$$

This expression can be extracted from a molecular dynamics simulation, since both the per-atom stress and the stress of the entire simulation box is readily available. In the

canonical ensemble (NVT), the change of internal energy will be:

$$\Delta U = W + T\Delta S \tag{8.9}$$

When expanding the system, the entropy per temperature will increase, as more microstates become available. This has to be compensated by adding heat. In molecular dynamics, the thermostat effectively adds energy as entropy during simulation box expansion. Furthermore, this energy cannot increase the kinetic energy, since N and T are kept fixed, so all the added heat must be absorbed as *potential* energy. This means that expanding the simulation box adds a lot of potential energy to the system, both directly through mechanical work, and indirectly through heat. Storage of the mechanical work is in line with the intuition: Particles are positioned higher in each others potentials, thus the potential energy increases. The heat absorption is more subtle: The system creates more available microstates when it expands homogeneously at a constant temperature. These microstates do only exist because of the strained state of the system, and will disappear if the strain disappears, for example during fracture. The entropy energy is not available for mechanical work. This is why Helmholtz free energy should be considered:

$$F = U - TS \tag{8.10}$$

This is exactly the energy available for mechanical work. The reason for this tedious discussion of expanding a box is that Helmholtz free energy is not directly available from molecular dynamics simulations. The change in Helmholtz free energy must be explicitly calculated by integrating the stress tensor of the system with incremental box deformations. When analyzing fracture, it is tempting to use the change in total energy or potential energy to calculate the fracture toughness. But this would be wrong – the fracture toughness should be calculated using the available mechanical energy. The energy release rate is the amount of Helmholtz free energy needed to create the projected crack surface area, not the change in potential energy or total energy of the system during fracture. This distinction is not easy to see when reading fracture mechanics, as fracture mechanics usually deal with elastic bodies with no temperature, but it is crucial when studying fracture in molecular dynamics.

When the crack propagates, the stress in the methane hydrate is released, and the system consists of a piece of methane hydrate, and some free gas in between. The entropy of this system is unknown, which means that calculating the surface energy becomes complicated. I have not gone further in how to calculate the surface energy, and assume that the entropic energy contribution by the free methane is small compared to the mechanical energy contributed by the crack surface opening. I find some support to this assumption, in the fact that the final potential energy in the proof-of-concept runs with different final strains were seemingly equal.

Hantal *et al.* [30] proposes to calculate the critical energy release rate with the following formula (This method was used for illite with ClayFF and ReaxFF):

$$\mathcal{G}_c = V \frac{\int_0^{E_{\text{final}}} \underline{\underline{\Sigma}}(E) : \mathrm{d}\underline{\underline{E}}}{\int_0^{E_{\text{final}}} \mathrm{d}A_{crack}} \tag{8.11}$$

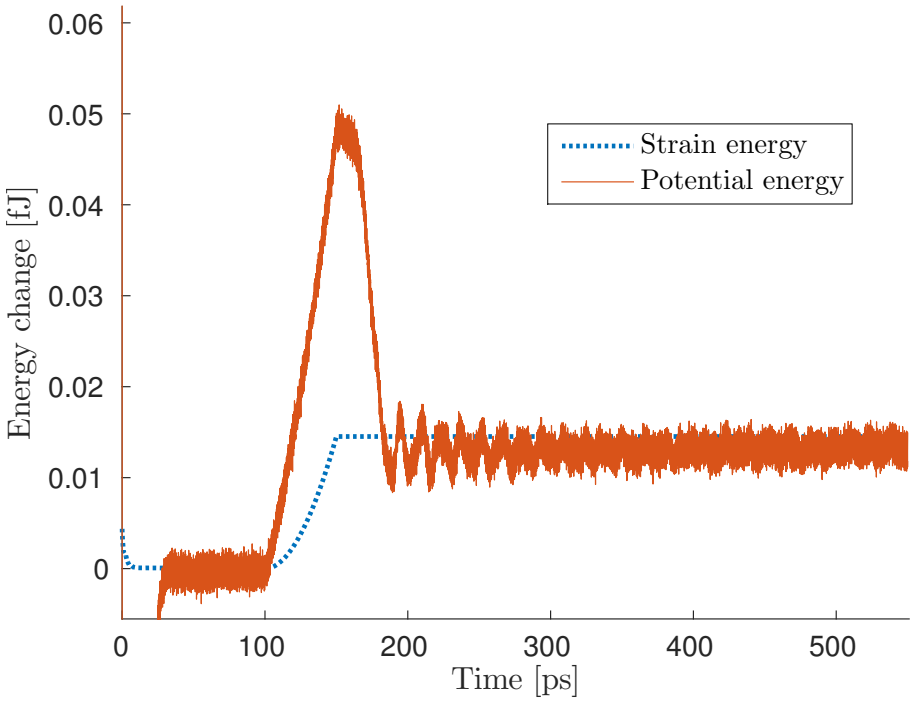


Figure 8.11: Potential energy and strain energy from a simulation where the strain before crack propagation is 0.048. The strain energy difference from before to after straining is 0.014 55 fJ. It is clear from this figure that the entropic energy introduced during straining is higher than the strain energy.

This is in principle sufficient to determine the critical energy release rate for any mode of loading For expansion of the simulation box along the x-axis, the strain integral (nominator) is equivalent to equation 8.8. The area integral (denominator) is just the crack area in time, which can be estimated using the crack tracer I described in chapter 6. For reference, Hantal *et al.* [30] also used a variation of the crack tracer algorithm that I have chosen to use.

Using equation 8.11 to measure the strain energy, I can recalculate the fracture toughness. I run a series of new simulations to be able to pinpoint the strain level at which the hydrate fails. Figure 8.11 shows the potential energy *and* the strain energy from a simulation where the strain before crack propagation was 0.048. It is clear from the figure that the entropic energy contribution (which is supplied by the thermostat) is larger than the strain energy, which means the total potential energy gives a bad estimate of the free energy, and ultimately, gives a bad estimate of the fracture toughness. This particular simulation is chosen because it is the simulation with the lowest required strain level to support a crack in a series of simulations with a strain increment of 0.001. The strain energy in this simulation is 0.014 55 fJ. The cross-sectional area of the simulation box is 12.04 Å × 288.8 Å, so my current best estimate of the critical energy release rate is $\mathcal{G}_c = 0.42\,\mathrm{J\,m^{-2}}$. This corresponds to a fracture toughness of $K_{Ic} = 0.060\ \mathrm{MPam^{\frac{1}{2}}}$.

8.5 Large simulations of fracture

Larger simulations allow for statistically more robust results. During crack initiation in the previous simulations, only a few particles near the crack tip ended up being essential to the crack initiation. Even though the periodic boundary condition allows for the crack to be thought of as infinitely deep, the low number of particles makes the crack initiation vulnerable to statistical fluctuations. With thicker systems, more particles reside in the region near the crack tip, resulting in better statistics. An obvious drawback of larger simulations is the added computational cost. The largest simulations I will perform systematically will be systems of $24 \times 24 \times 12$ sI unit cells over 550 ps. The computational cost of each of these simulation is $\sim$ 6000 CPU-hours, taking the total computational cost of this work to around 10^5 CPU-hours. Also, the amounts of data increase drastically if per-particle properties are to be stored. My approach has been to post-process trajectory data from simulations, which means I have had to deal with large amounts of data. This has been a challenge when working with large simulations, and has required close attention to efficiency in file handling, both to reduce the CPU-time of analyzes, and to keep the memory demands on an acceptable level. Also, the amount of data that can be output is limited because of limited amounts of storage space. Outputting all positions and velocities in a simulation of 10^6 particles, which is the number of particles in my largest simulations, requires around 50 MB per timestep. That becomes 50 TB ns^{-1} for a timestep of 1 fs.

8.5.1 Complete analysis of a single simulation

In the following, I use the analysis tools that were presented in the tools section to analyze *one* simulation. This simulation will be referred to as the *full-analysis simulation*. The analyzes here will be representative of how data is obtained when showing up in later results where data from several simulations are used together to say something more general.

Below follows simulation details. Some parameters are not mentioned in this description. They are the same as the ones listed for the proof-of-concept simulations in table 8.2.

Simulation details: A system consisting of $24 \times 24 \times 12$ sI unit cells is prepared. First, an elliptical hole in the xy-plane spanning the whole z-direction is carved out as described in section 6.2. The system is then allowed to equilibrate with an anisotropic NPT thermo-barostat for 100 ps, with ambient barostat pressure (101 325 Pa). The system is then integrated NVT, but subjected to a constant strain rate taking it to a strain level of $\epsilon_{xx} = 0.048$ after 50 ps. Then straining is stopped, and the system is left NVT for 350 ps. The damping time of the thermostat and barostat during the whole simulations is $t_{damp} = 1$ ps. Additionally, a drag coefficient of 1.0 (as described in the LAMMPS documentation) is added, as this is recommended to damp oscillations in solids. The thermostat temperature is set to 260 K. The parameters applied here, are the same as in the proof-of-concept simulations, with the exception that it is thicker, and that the simulation time is

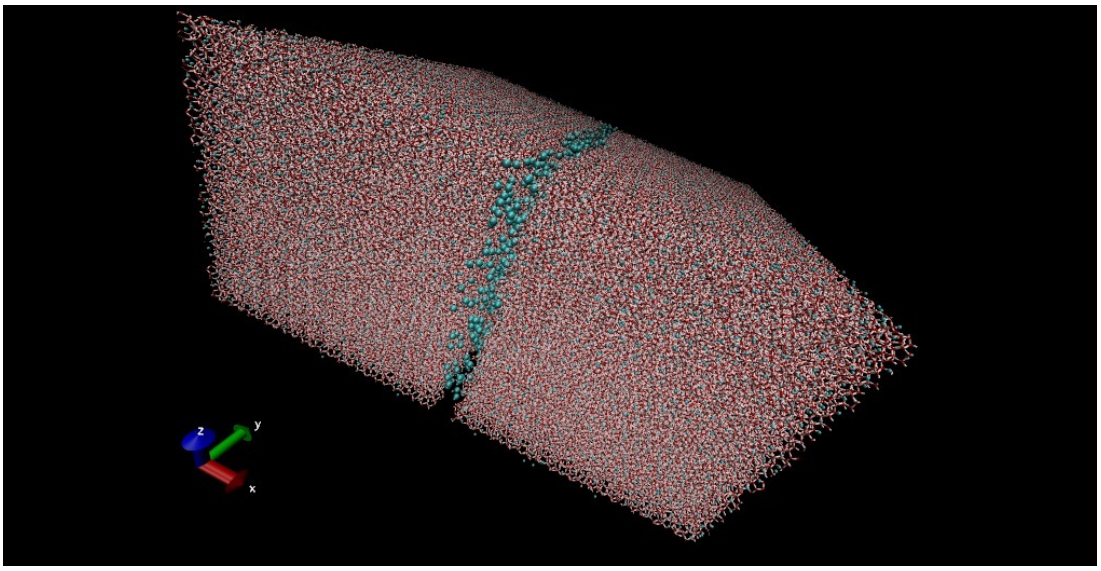


Figure 8.12: Perspective image of a representative state after fracture in a 24×24×12 sI unit cell simulation. The biggest green spheres are methane molecules that are considered free to move (not enclathrated). Smaller green spheres (barely visible) are enclathrated methane molecules. Water molecules are drawn as covalent OH-bonds and hydrogen bonds.

slightly shorter.

This simulation features a crack spanning the whole yz-plane. Figure 8.12 shows what the system looks like in perspective after crack propagation. The other pictures are rendered with an orthographic projection, so the system will look two-dimensional.

In this simulation, it takes around 30 ps from the system is fully strained until a crack starts propagating. Figure 8.13 shows pictures of the crack evolution in time, to give an idea of what the system looks like during fracture. It can be seen that the crack is very straight.

Figure 8.14 shows the potential energy, the kinetic energy and the strain energy in time for the whole simulation. It shows that a strain of 0.048 results in a strain energy only slightly higher than what is needed if the criterion for a crack to propagate is $\mathcal{G}_c = 2\gamma_s$. This means the system shows brittle behavior in this particular simulation. The kinetic energy slightly decreases during straining. This is the reason why the thermostat supplies energy to the system, as was discussed in section 8.4. During fracture, the kinetic energy increases, which means the temperature increases during fracture. The heat supplied by the thermostat during straining is now released. The reason why the temperature change is visible, is that there is a finite damping time on the thermostat, so that the energy corrections are not immediate. The temperature is not indicated in the figure, but the temperature changes are only by a few Kelvin.

Figure 8.15 shows the measured crack area in time, and its time derivative which estimates the crack tip speed. The maximum crack tip speed is around $1\,\mathrm{km\,s^{-1}}$. A little

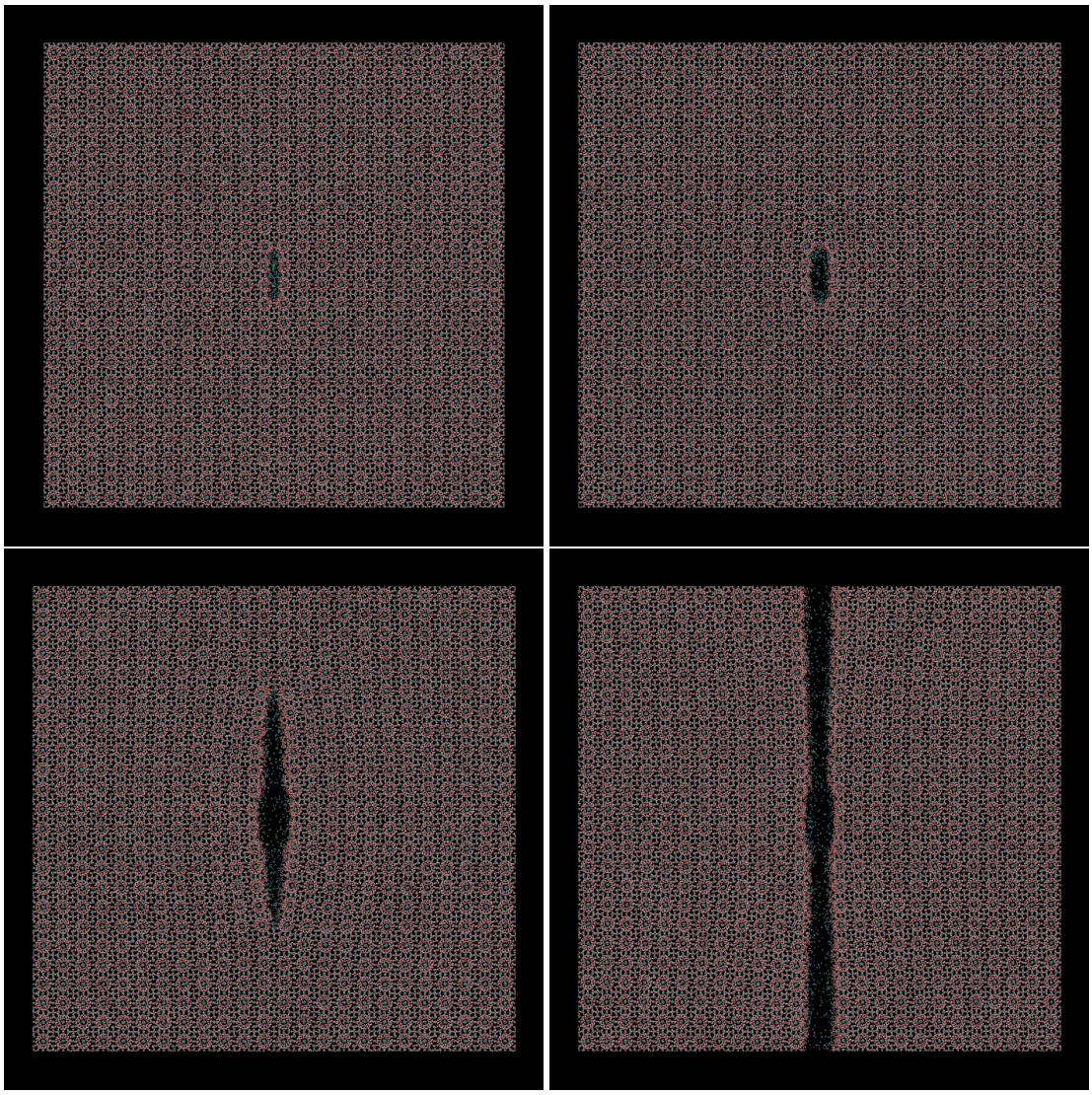


Figure 8.13: Evolution of the crack in the full-analysis simulation in time: Equilibrated system at 90 ps (upper left). Fully strained system ($\varepsilon_{xx} = 0.048$) at 150 ps (upper right). Crack has started propagating at 180 ps (lower left). Crack is fully propagated after 210 ps (lower right). The crack is wider near the periodic boundary in the y-direction in the last frame due to global oscillations following the crack propagation.

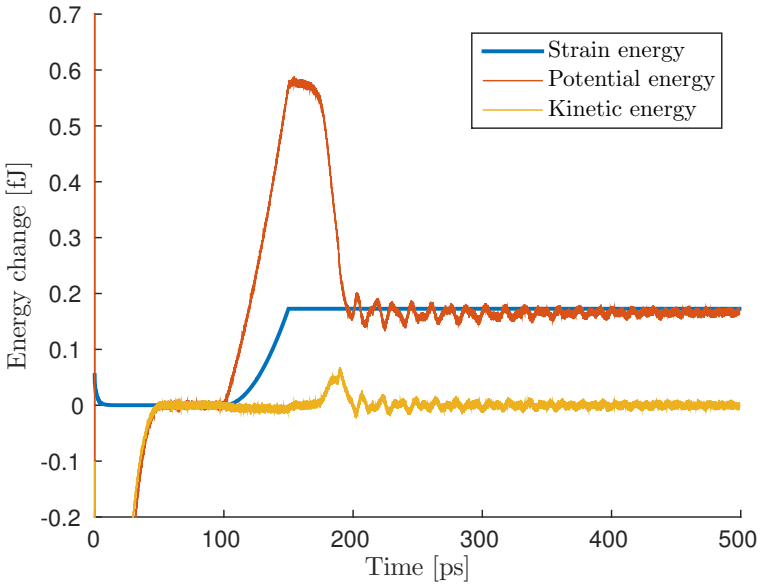


Figure 8.14: Strain energy, potential energy and kinetic energy in the full-analysis simulation. The values are rebased at $t = 100\,\text{ps}$ where straining starts. The kinetic energy corresponds to the temperature, and we see that the temperature slightly decreases while the strain increases, and increases during crack propagation. The strain energy imposed on the system is just slightly higher than the energy difference between the unstrained system and the fractured system.

bump can be seen in the crack tip speed during crack propagation. This is a feature that has been seen in all the crack tip speed measurements in this work. My guess is that it is caused by bulk elastic waves triggered by the crack. The fluctuations of the measurement in figure 8.15 (full-analysis simulation) are significantly lower than the fluctuations for the same measure in figure 8.9, which shows data from a thin system. The fluctuations are best seen in the crack tip speed plots, as differentiation enhances noise. The parameters for the crack analysis were the same in both these measurements: $N = 10^6$, $\Delta l = 1\,\text{Å}$ and $r_p = 4.0\,\text{Å}$. Therefore, the fluctuations in the measured crack area must be due to the fluctuations in the underlying solvent-accessible surface area (which are relatively larger for the thinner system), not due to too poor sampling of the surface.

Figure 8.16 shows all components of the stress tensor of the system in space during the period when the system is fully strained, but before propagation starts. It is averaged over the z-direction in space and over 20 ps in time to get statistics for a nice picture. The stress field is visually compatible with the analytical solution close to the crack tip for a homogeneous isotropic linear elastic solid. The shear components normal to the xy-plane are small, but not negligible, which means that the plane strain condition is only partially met. Particularly, the amplitude of τ_{xz} is about 1/5 of that of τ_{xy}.

Figure 8.17 is similar to figure 8.16, but it now shows the stress field *during* crack propagation. Since tracking the crack requires higher time resolution, the stresses are only averaged over 0.3 ps. As expected, the region of high stresses moves along with

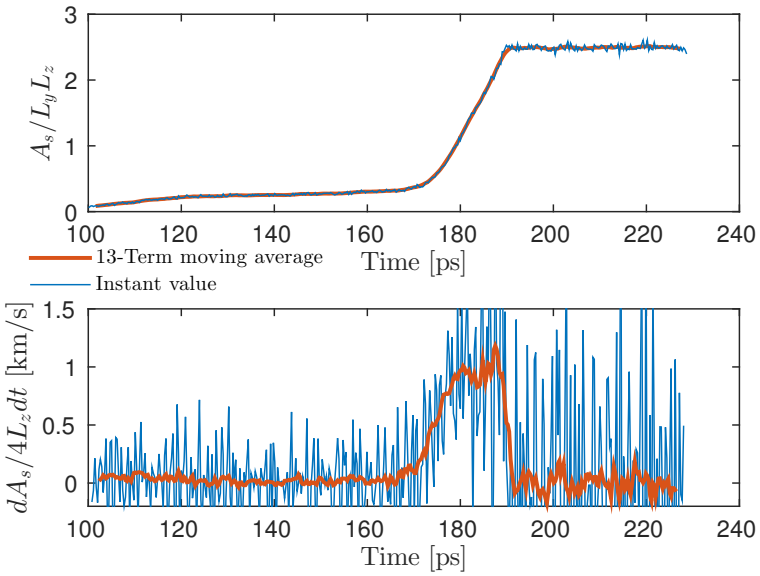


Figure 8.15: Crack surface area evolution in time. The upper panel shows the measured area, and the lower panel is an estimate of the crack tip speed. The derivative for the lower panel is calculated using a standard 5-point stencil for the first derivative. The crack surface area was measured using the Monte Carlo procedure described earlier, with $N_s = 10^6$, $r_p = 4.0\,\text{Å}$ and $\Delta l = 1.0\,\text{Å}$. Note the little bump in the crack speed around 180 ps.

the crack tip.

Figure 8.18 shows the stress σ_{xx} at several points in time before, during and after crack propagation. Here also, the stresses are only averaged over 0.3 ps. The top crack is leading the bottom crack with $\sim 10\,\text{Å}$, which is probably coincidental

The stress-field analyzes are at this point only useful to get an idea of what the stress-field looks like, and I will not follow up with comparisons of the stress-fields between different simulations.

Figure 8.19 shows the system after fracture with the free methane – methane that was released as a consequence of the crack – indicated as big spheres. Free methane characterization is explained in the next section, and represents a small detour before going on to comparing results between simulations with different strains and temperatures, to make more general analyzes of fracture in methane hydrates.

8.6 A simple characterization of the fracture: The amount of free methane.

I believe that the amount of free methane can be a valuable measure of how the crack surface looks because free methane no longer supports a cage structure. Water molecules either stay on the crack surface or are immediately drawn towards the crack surface in a matter of picoseconds after a crack has passed. Methane, on the other hand, is free to move in the pore space created by the crack. A first order approach to crack

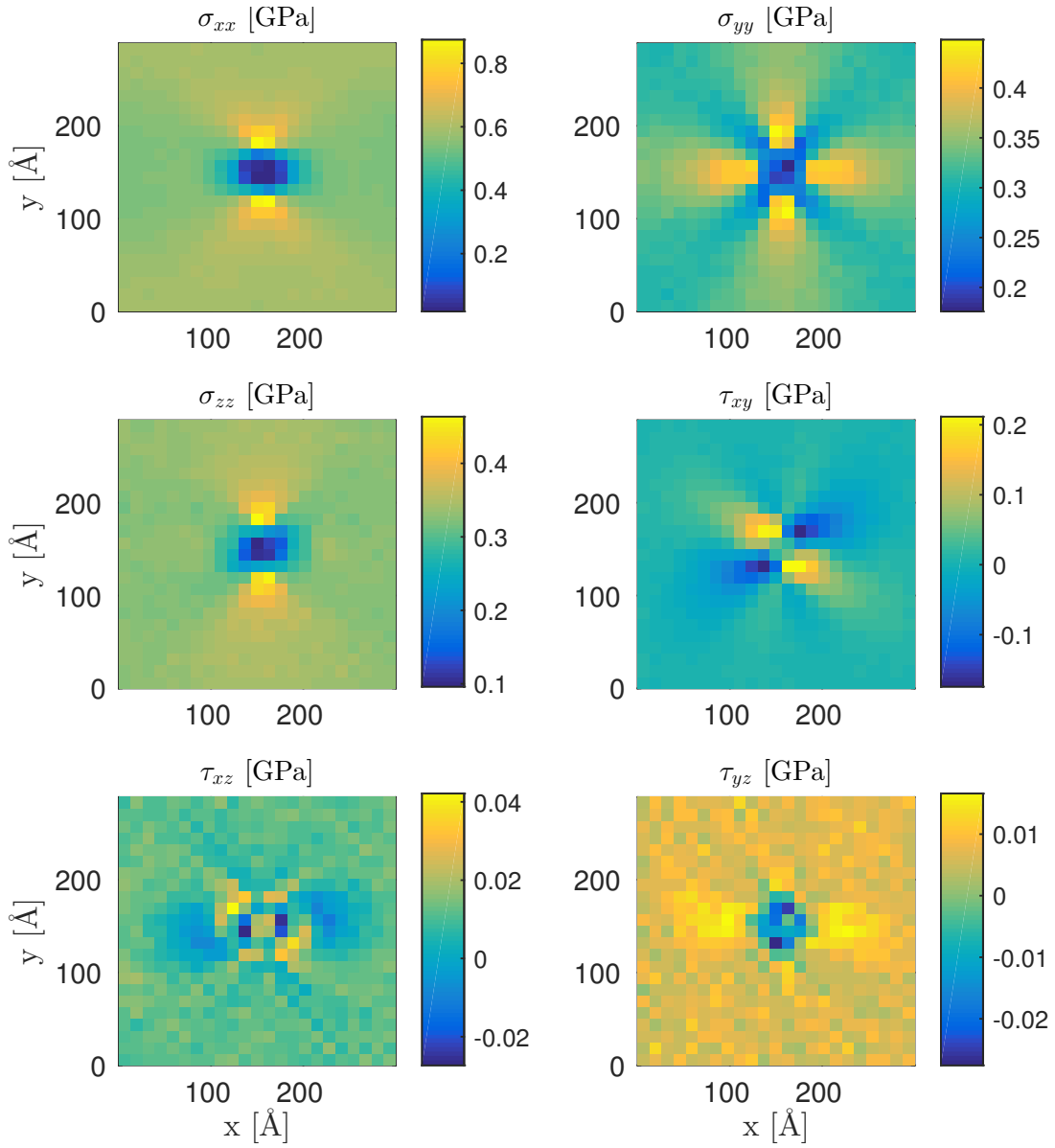


Figure 8.16: Stress field averaged over 20 ps just after straining was stopped, and while waiting for a crack to start. The modeled system is $24 \times 24 \times 12$ sI unit cells. The four uppermost panels show features qualitatively similar to the analytical solution of the stress field around the crack tip of elliptical hole in a linearly elastic and isotropic material (see figure 3.4). The shear stress components outside the xy-plane are relatively small compared to the components in the xy-plane, which means the plane strain condition is satisfied fairly well. Note also that the stress σ_{zz} is similar to σ_{xx} due to L_z being fixed, which disallows Poisson contraction and results in a stress.

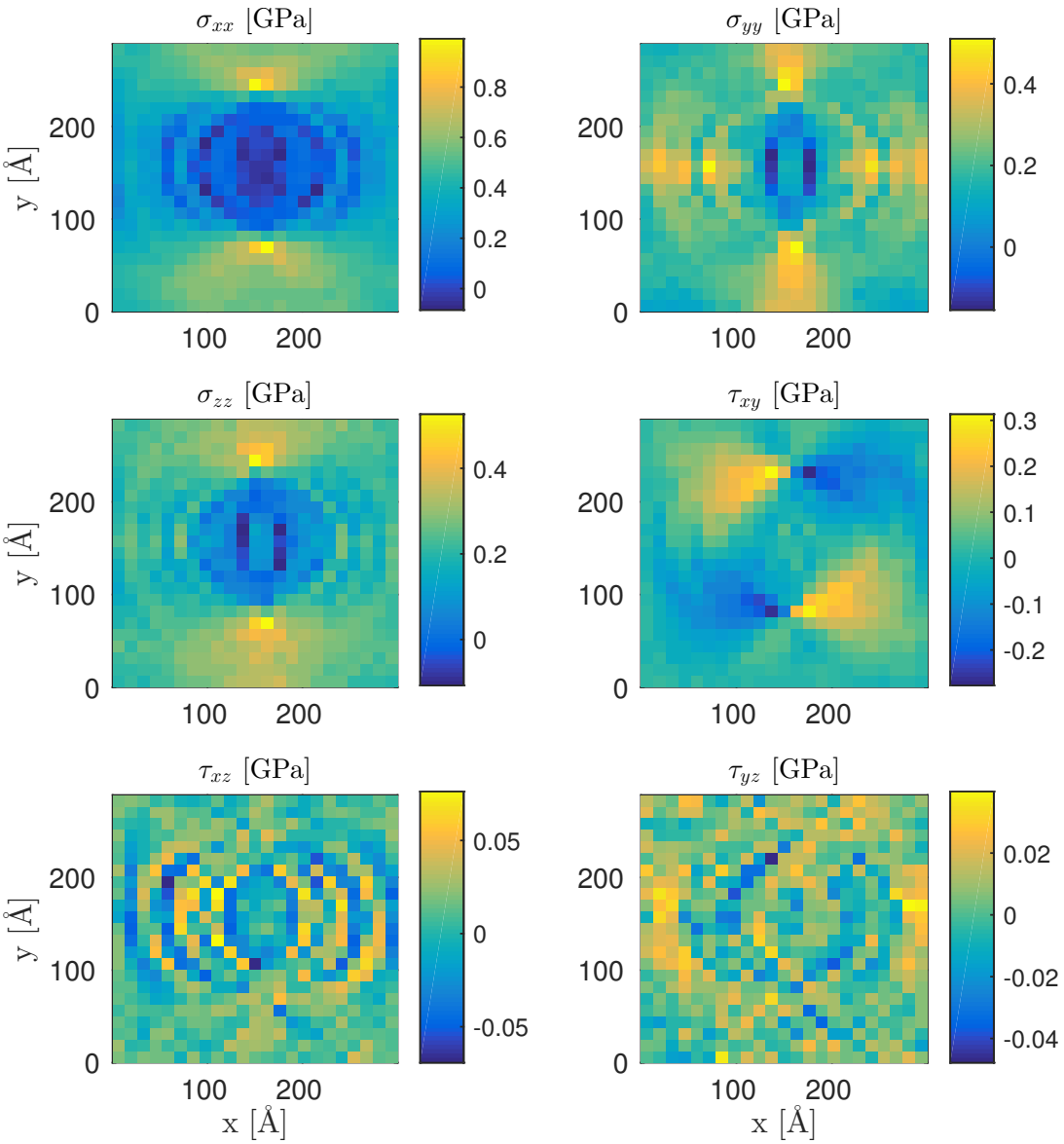


Figure 8.17: Stress field during crack propagation in the full-analysis simulation. Stresses are averaged over 0.3 ps before $t = 180$ ps.

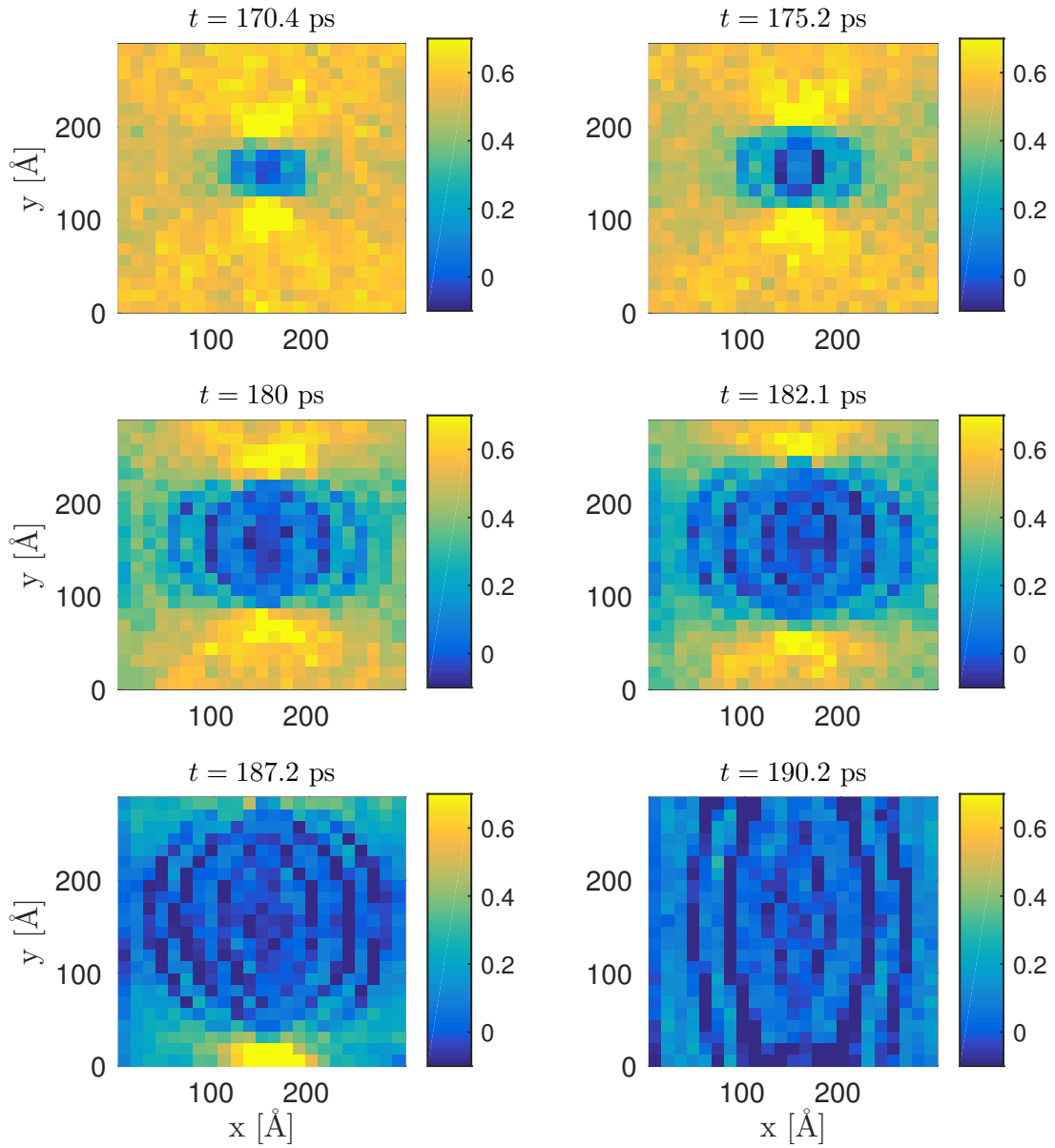


Figure 8.18: Time-lapse of the stress σ_{xx} during crack propagation. There are clear signs of sound waves, and the shape of the waves indicate that the crack is traveling at a speed on the same order of magnitude as the shear wave speed. The color scale is kept equal between the simulations for easier comparison. Because of the fixed color scale, the stress near the crack tip in some of the figures saturate the color scale. Notice that the two edges of the initial hole don't crack at the same time. The upper crack is leading the lower crack with $\sim 10\,\text{Å}$.

surface characterization is therefore to count free methane in the pore space and use that to estimate the amount of hydrate that was dissociated during fracture. The simplest estimate is obtained by using the hydrate number n_w (5.75 for fully occupied sI hydrate) to count the amount of water molecules that are no longer part of the stable hydrate structure. Or, equivalently, use that there are 8 methane molecules in each unit cell, and calculate the number of unit cells that no longer have their methane molecules, and may be considered dissociated.

8.6.1 Estimate by counting

A straightforward approach for calculating the amount of free methane, is to go through all methane positions, and check whether they are part of the wall or the void. This is similar to what was done when tracking cracks, only that random points, not methane molecules, were checked for whether they were part of the void or the wall. In this method, I choose to consider a methane molecule to be free if it was part of the void at any time during the simulation. The reason for the "at any time during the simulation" is that measuring frame by frame largely underestimates the number of free methane molecules. Methane molecules that are free are not necessarily far enough from the closest water molecule to be considered as part of the void, but they *are* free to move. To check that the free methane molecules are indeed free, the positions of the methane molecules that were defined as void can be visualized at a point in time after the crack has propagated. Since free methane is accumulated in the measurement, a slightly higher r_p is needed than for crack tracing – near the lower reasonable limit for r_p, some improbable (but they will occur in large simulations) wall configurations will result in wall methane being counted as free. Figure 8.19 is a snapshot of the system with the free methane molecules highlighted. Some of the methane molecules that are considered free actually reside in small cages near the fracture surface, and it is not obvious whether they should be considered free or not. I tend towards considering them to be free, since they have proven that they are in cages that are not sufficiently intact to keep the methane molecules over time. Most of the free methane molecules are clearly free, so this should not be a big issue – but it can be interesting to study for instance the rate at which methane is captured and released by the near-crack-surface cages.

8.6.2 Results on free methane

For the analysis of free methane, I choose to perform a series of simulations with double the thickness of the thinnest system, to be able to see whether this can affect the amount of methane that is freed before, during and after crack propagation. Figure 8.20 shows the amount of free methane during simulation for the two different thicknesses of the system, $L_z \approx 12\,\text{Å}$ and $L_z \approx 24\,\text{Å}$. The number of free methane molecules is rescaled using the number of sI unit cells per cross section parallel to the crack. With full occupancy, the number of methane molecules per sI cell is 8, and if a whole plane of unit cells released their methane molecules, that would yield a number of 8 free methane molecules after rescaling. The number of free methane molecules shows no significant variance with the system thickness in these simulations. This is a bit surprising, since the crack propagation has seemed more jittery for thinner systems, which could have

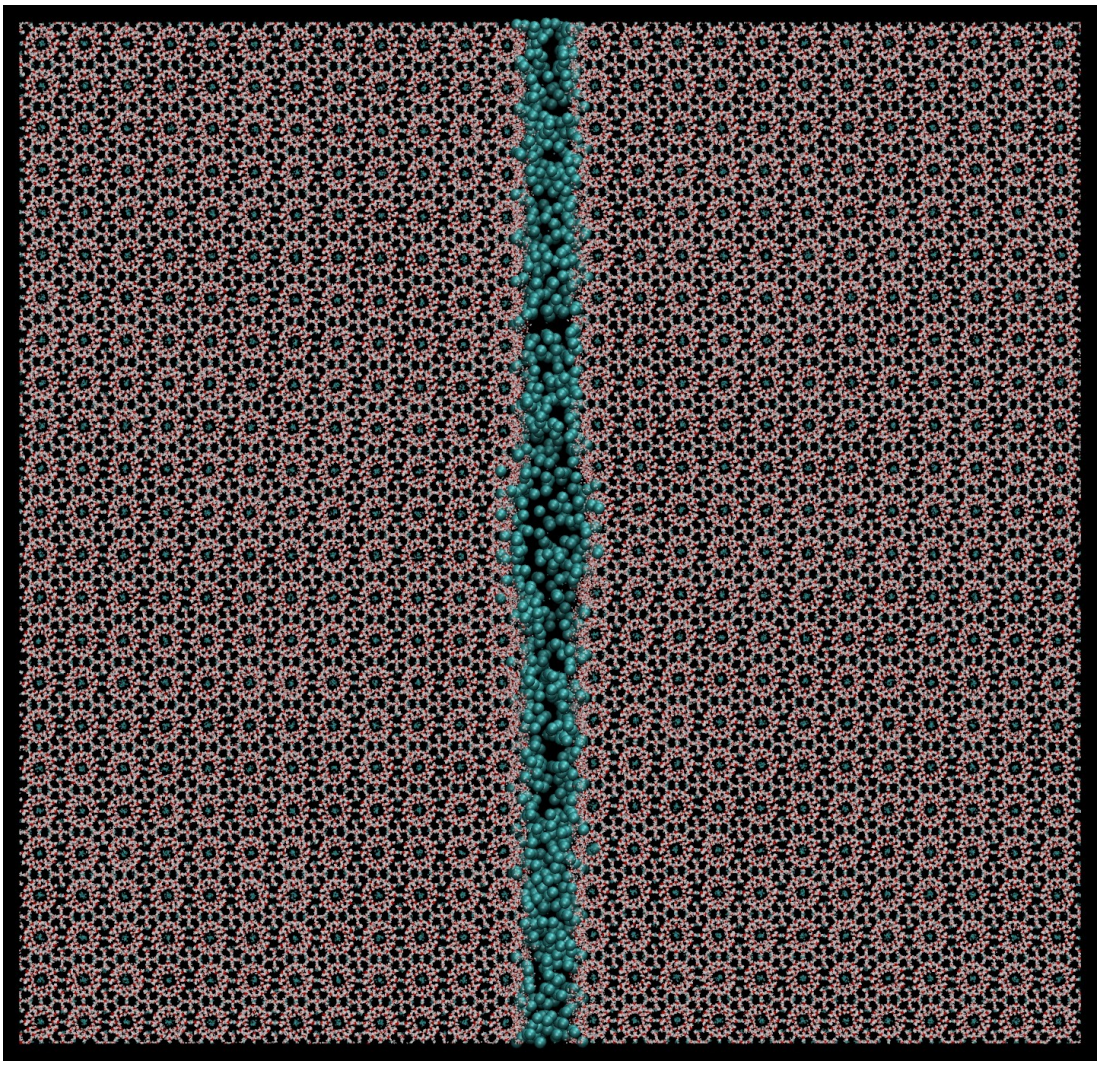


Figure 8.19: Free methane – particles that were more than $r_p = 4.5\,\text{Å}$ away from any water molecule at some point – are drawn as big green spheres. Other methane molecules are barely visible particles. Note that some "free" methane molecules actually occupy cages. Most of them are in the small cages nearest to the crack, but a few also occupy big cages, that are a bit farther from the crack surface.

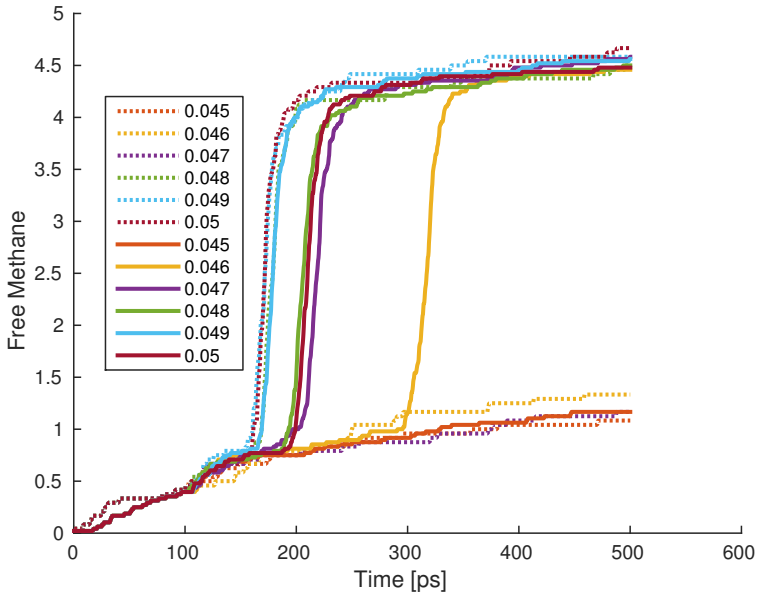


Figure 8.20: Free methane in time for systems with $L_z \approx 12\,\text{Å}$ (dashed lines) $L_z \approx 24\,\text{Å}$ (solid lines) for different values of the final strain (legend). Free methane is normalized by the number of unit cells in the fracture plane, so the plotted value is: Free Methane Molecules $\cdot\, L_{sI}^2/(L_y L_z)$.

led to more methane being released. There might still be an effect, but in that case the effect must be small.

8.6.3 Diffusion in the crack space

Before I decided that counting the number of free methane molecules was a suitably good method to study free methane, I explored an indirect way of measuring the amount of free methane. The idea was to assume the motion of the free methane to be diffusive. The diffusion constant could then be measured and compared to what it would have been if all methane molecules were free to move. Unfortunately, the diffusion constant varies too much with the density of the methane gas for this to be a viable method – both the number of methane molecules and the volume of the pore space is unknown. Furthermore, diffusion is not expected to be the same in pores as in a bulk Lennard-Jones fluid (see e.g. Pozhar [58, p. 18]). Therefore, this method was discarded. An additional problem is the global oscillations after crack propagation, which might further disturb the diffusion of particles.

Even though measuring the number of free methanes by mean squared displacement turned out to be difficult, the diffusion constant of the free methane can still be interesting to measure. Knowing the number of free methane molecules, and assuming that the rest of the methanes have exhausted their potential of contributing to the mean squared displacement, the diffusion constant in the pore space can be measured. Figure 8.21 shows the per-particle mean squared displacement of methane and water in the full-analysis simulation. The slope of the mean squared displacement is calculated to

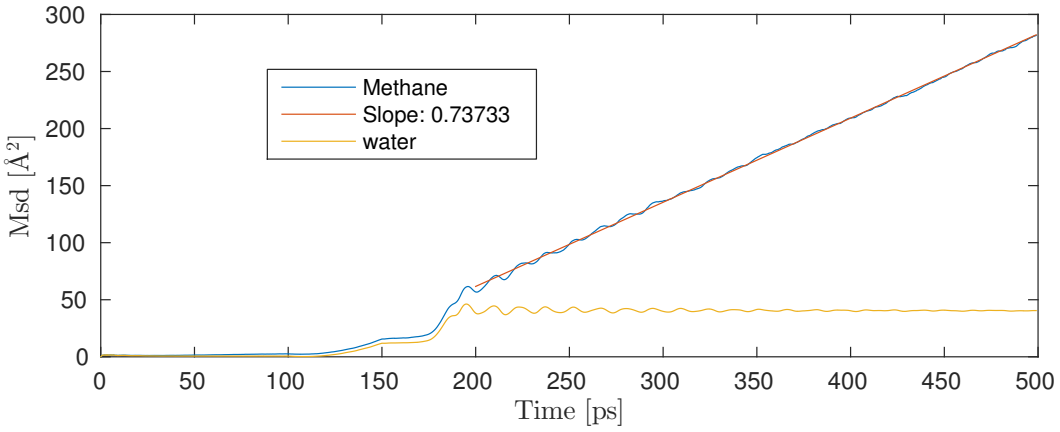


Figure 8.21: Per-particle mean squared displacement of methane and water in the full-analysis simulation. The slope of the methane mean squared displacement is $0.74\,\text{Å}^2\,\text{ps}^{-1}$.

$0.74\,\text{Å}^2\,\text{ps}^{-1}$, which corresponds to a diffusion coefficient of $D_E = 1.2 \times 10^{-9}\,\text{m}^2\,\text{s}^{-1}$ by using the Einstein relation. That value makes little sense, as it includes the enclathrated methane. When correcting for the fraction of methane molecules that are considered free to move, in this simulation it is 2.4 %, the diffusion coefficient of the free methane becomes $D_E = 5.3 \times 10^{-8}\,\text{m}^2\,\text{s}^{-1}$. This value is probably an underestimation of the diffusion coefficient for the particles that are actually diffusing, as many methanes are trapped by the wall. What can be said, however, is that this is a very high diffusion coefficient, which, given the temperature, corresponds to a low pressure. A possible simulation to compare with is a simulation by Cao & Wu [16] of Lennard-Jones methane in carbon nanotubes. At a temperature of 267 K, they found self-diffusion coefficients for methane of around $1 \times 10^{-8}\,\text{m}^2\,\text{s}^{-1}$ to $10 \times 10^{-8}\,\text{m}^2\,\text{s}^{-1}$ for pressures ranging from 1 MPa to 8 MPa. The simulations were done in carbon nanotubes with diameters from 20 Å to 40 Å. This comparison is far-fetched, but it is reassuring that the results are in the right ballpark. It should be noted that the methane I observe in the crack is very dilute, possibly to the extent where Knudsen diffusion is a reasonable approximation. The methane being dilute is not obvious from the pictures of the system, since methanes that are behind each other seem to crowd the crack. If looking at the thinner system, with $L_z \approx 12\,\text{Å}$, the diluteness becomes clearer (see figure 8.7).

8.7 Increasing the temperature

All simulations until now have been run at $T = 260\,\text{K}$. This is a relatively low temperature, and specifically, it is lower than the melting point of water for the TIP4P/Ice water model. I now want to find out what happens when the temperature is higher than the melting point of water in the model I apply. I will run a few simulations in the large system, $24 \times 24 \times 12$ sI unit cells, and report results. Based on these results, I will decide whether a further investigation of temperature changes is warranted.

The new simulations are run at a temperature 280 K, which is higher than the

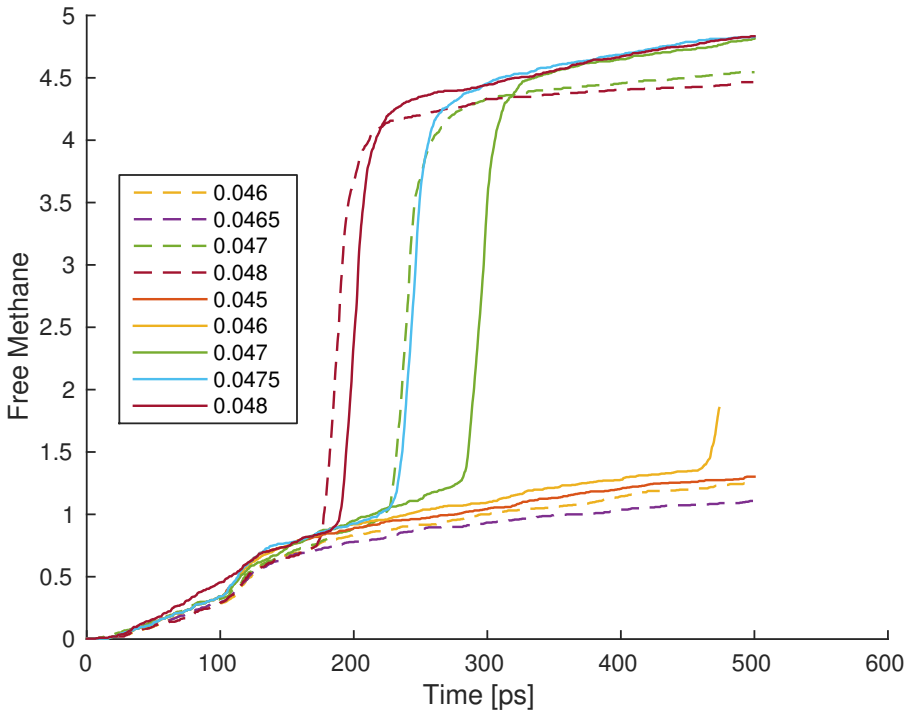


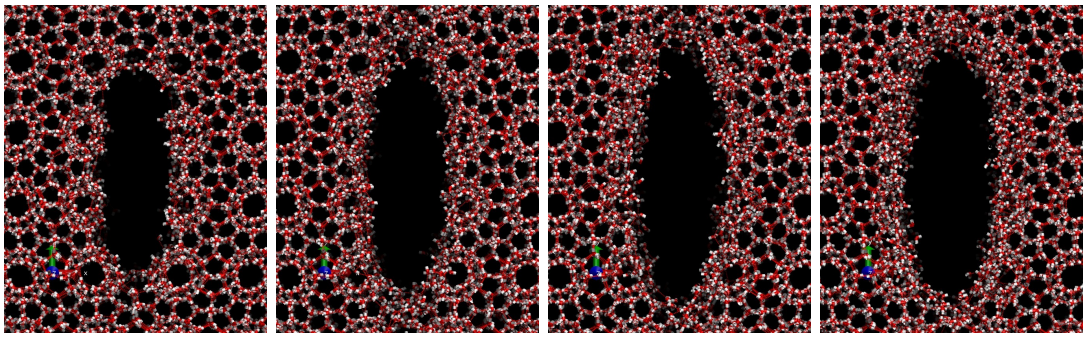
Figure 8.22: Released methane for different strain levels (legend) with $T = 260\,\mathrm{K}$ (dashed lines) and $T = 280\,\mathrm{K}$ (solid lines). The amount of free methane gets significantly higher with the increased temperature.

melting point of the TIP4P/Ice water model.

Since the total crack area is limited by the cross-sectional area of the simulation box, I choose to look at the amount of free methane in the simulations with $T = 280\,\mathrm{K}$, and to compare it with the corresponding data for $T = 260\,\mathrm{K}$. Figure 8.22 shows the amount of free methane in simulations with varying values of the strain level, and the two different temperatures. It is clear that the amount of free methane becomes higher when the temperature is set higher, but it is unclear whether the amount of methane that is freed *during* fracture is different. The clearest tendency is that methane is freed more quickly before and after crack propagation when the temperature is higher. This points towards the temperature playing a role for slow processes related to fracture. This will be further discussed and analyzed later. A larger study of varying temperatures will be considered in the future.

8.8 Investigating simulations with no system-spanning crack

All simulations of methane hydrates with cracks up to now, except for the simulation with a cavity in front of the crack, has had a total simulation time of 500–550 ps. Since there is a waiting time before fracture starts, there might be a potential of having fracture in the non-fractured systems if the simulation is allowed to last for sufficiently long. It is also possible that something else will happen. The crack can for example



(a) $t = 150\,\text{ps}$ **(b)** $t = 300\,\text{ps}$ **(c)** $t = 600\,\text{ps}$ **(d)** $t = 900\,\text{ps}$

Figure 8.23: Images zoomed in on the initial crack at different points in time in a long simulation. Methane molecules are hidden for a better visual experience of the water. No global crack propagates, but there is still development of the small initial crack. From $t = 150\,\text{ps}$ to $t = 900\,\text{ps}$ the water-rich region at the wall–void interface is getting less ordered, and after $t = 900\,\text{ps}$ there is a water film between the sI hydrate and the free methane in the crack.

change its shape, strengthening the sample – this is necessary for the material to have ductile properties. It is also possible that a much slower crack can propagate, possibly not reaching the periodic boundary if the strain does not contribute enough energy to open a surface spanning the whole yz-plane.

In this section, I look at a simulation very similar to the one described in section 8.5. The only difference is that the strain level after 150 ps is 0.045, and that the temperature is 280 K. No system spanning crack is observed during the 400 ps after straining. There is, however, a small increase in the surface area, as can be seen from the orange dotted line in the lower panel of figure 8.25.

Since the system does not seem to have equilibrated (the potential energy is still falling), I continue the simulation for another 400 ps, to see what happens. Figure 8.23 shows images of the crack at selected points in time, and shows that even though the crack does not span the system after the simulations, it *does* develop. It seems like the system is trying to minimize its mechanical energy by creating an optimal surface – it looks like the crack is trying to optimize its shape. This is probably a different process than the process leading to rapid crack initiation and propagation. It can for instance be that the process can bee usefully described with a thermal activation and a stress activation.

8.9 Aggregated results

In this section, I present overall tendencies that arise from putting together the results I have already presented with results from additional simulations of the same kind, but with different strain levels. I end the section with a comparison of my results with the theory of linear elastic fracture mechanics.

This section contains three figures: Figure 8.24 shows the measured surface area, the derived crack tip speed and the tensile stress on a selection of simulations. All

simulations shown in this figure are of the thickest type, $L_z \approx 144\,\text{Å}$, and data are shown for both $T = 260\,\text{K}$ and $T = 280\,\text{K}$. Figure 8.25 is a zoom of figure 8.24. Figure 8.26 uses the data from figure 8.24, but plots the tensile stress against the crack length (which is derived from the crack area). This is efficient for comparison with linear elastic fracture mechanics.

8.9.1 Effects of the system thickness

The study of finite size effects has not yet gotten much attention in this work, but from visual inspection of simulations with different thicknesses L_z, it is clear that there are finite thickness effects. Until now, three system sizes has been studied: A thickness of one sI unit cell, two sI unit cells, and a thickness of 12 sI unit cells. All systems had 24 unit cells in the x- and y-directions.

Most notably, the crack propagation in the thickest system seems more predictable than that in the thinnest system. Particularly, the crack tip in the small simulations can change direction for a short time, and then continue propagating in the y-direction but in another column of unit cells than it did before. Similar behavior has not been observed in the large system.

To systematically study this effect, I have compared the crack surface area in simulations with systems of different thicknesses, but with the same strain rates. A chaotic crack should leave a larger crack surface area than a straight one.

The result is that the system thickness does not change the relative crack area; potential differences are masked by the fluctuations in the measured crack surface area. I do not show a figure for this.

8.9.2 Waiting time for fracture

Waiting times before fracture can be read from figure 8.24 by looking at when the area or crack speed suddenly increases. In the simulations with $T = 260\,\text{K}$, the waiting time is short for the highly strained samples, and the samples subjected to too little strain do not fracture. Between these extremes, there is not a general rule that higher strains give shorter waiting times. The simulations with $T = 280\,\text{K}$, on the other hand, shows more systematic behavior. A simulation with a higher strain systematically has a shorter waiting time before fracture in these systems. It is possible that the uncertainty in the waiting time before fracture at a given strain level becomes smaller when temperature increases, and especially when exceeding the melting temperature of pure water in the given model.

A possible explanation for such a scenario is the following: Consider melting to be a process constructed by many small events. A high temperature will yield a high frequency of such events, and a low temperature will yield a low frequency. During a given amount of time, a system with high temperature will melt more than a system with low temperature, but additionally, since the melting events are considered random, the relative uncertainty of the melted amount will be lower for the high-temperature system. If it is now assumed that two systems are prepared: The first with a high temperature and low strain, and the second with a low temperature and high strain, but so that the expected time to wait before a sufficient amount of water has melted

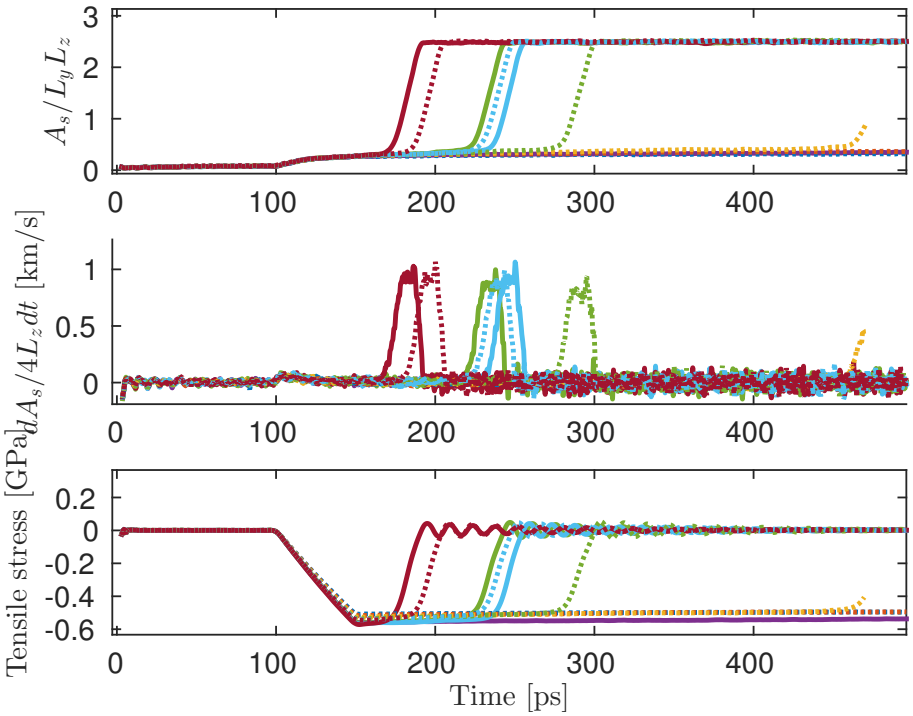


Figure 8.24: Crack area (top panel), crack speed (middle panel) and tensile stress (lower panel). Simulation temperatures are 260 K (solid lines) and 280 K (dashed lines). Strains: 0.044 (blue) 0.045 (orange), 0.046 (yellow), 0.0465 (violet), 0.047 (green), 0.0475 (cyan), 0.048 (cardinal red). The colors are the same as in the legend of figure 8.26. The crack areas and the stresses are shown as a 21-point moving average, and the crack speed is calculated from the moving average of the crack area.

to facilitate rapid cracks is equal for both systems. Then, the absolute uncertainty in the waiting time of the system with the higher temperature will be lower than that of the system with the lower temperature.

8.9.3 Crack area and crack tip speed

Crack areas and crack tip speeds have been measured both in the thin and the thick systems, but I have only made comparisons between different simulations for the thickest system. The measured crack area, crack speed and stress in the thick systems are shown both for $T = 260\,\mathrm{K}$ and $T = 280\,\mathrm{K}$ in figure 8.24. The crack area of the fractured system is essentially the same for all simulations where fracture occurs. The fracture *speed*, however, varies between the simulations: It is a clear tendency that the crack tip speed upon rupture increases with increasing strain level before crack propagation. This is unsurprising. The low accuracy of the speed measure makes is hard to make quantitative comparisons, and a longer system is probably needed for robust quantitative comparison. What can be said, is that even though the crack tip speed significantly depends on the strain, the differences in crack speed are not very large. In figure 8.25, all crack speeds fall roughly between $750\,\mathrm{m\,s^{-1}}$ and $1000\,\mathrm{m\,s^{-1}}$.

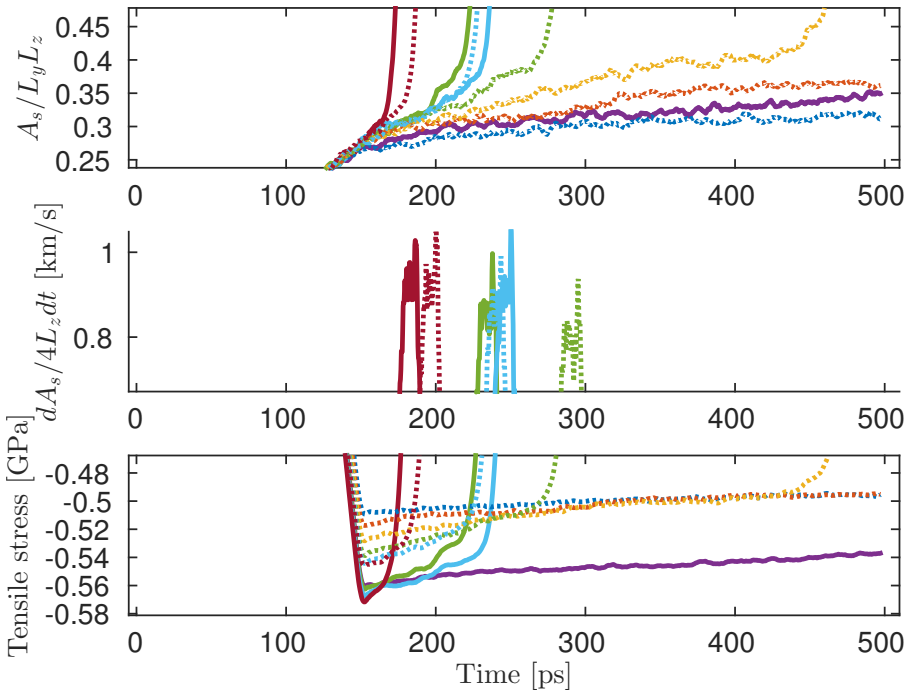


Figure 8.25: Crack area (top panel), crack speed (middle panel) and tensile stress (lower panel). The panels are the same as the ones in 8.24, but they are significantly zoomed. It becomes clear from the lower panel that a higher tensile stress (from a higher strain) before rupture results in a shorter waiting time before fracture in the simulations with $T = 280\,\mathrm{K}$ (dotted lines), but the same systematic effect is not seen for $T = 260\,\mathrm{K}$. Tendencies of two stages of crack growth can be seen from the upper panel. The area grows slowly until it suddenly starts to grow fast.

In terms of Rayleigh wave speeds, this range corresponds to $\approx 0.5 - 0.65 v_R$ (see table 8.1 for wave speeds in this model). This is well within the speed range where crack instabilities are expected to be present, but crack instabilities have not been prominent in the simulations in this work.

8.9.4 Free methane

Methane is freed almost instantly during crack propagation. This means that some methane hydrate is decomposed when the crack propagates. This can be seen directly by comparing figure 8.22 and figure 8.24. Also, the proportion of methane that is freed during rapid surface area growth is about the same as the proportion of surface area that is grown during rapid area growth. This further supports the statement that methane is freed quickly – it does not stay in the hydrate lattice and then diffuse out. It could have been different. For example, the crack could have split the methane hydrate without decomposing it, and then the methane molecules would have to diffuse out of the walls after crack propagation.

8.9.5 Surface energy and fracture toughness

The same exercise of measuring the critical energy release rate of the hydrate as in section 8.4 has been done for other system thicknesses and for the simulations with increased temperature. I have also read off the corresponding surface energy from figures like figure 8.11. Since the surface energy is fluctuating, these results are uncertain, but they can at least be used to say that the critical energy release rate is approximately two times the surface energy, which classifies the hydrate as brittle. The results are summarized in table 8.3, and they are all compatible with a fracture toughness of $0.058(2)\,\mathrm{MPam}^{\frac{1}{2}}$, where the uncertainty captures only the variation between the entries in table 8.3.

Since the surface energy in this work is gotten by simply checking the potential energy difference, it is unclear what it means. The free methane filling the crack can contribute energy that should probably rather be included in the crack-driving energy than in the crack-resisting energy, but I have not been investigating this possible effect systematically. Since the free methane + ice system has a lower density than the methane hydrate, methane is likely to contribute to driving cracks.

I stated that the surface energy measure was uncertain because of the fluctuations in the potential energy after fracture. However, a tendency that the surface energy I measure is higher than half the energy release rate is robust in the high-temperature simulations. This indicates that there is something else than the added strain energy that contributes to crack opening. It could also just be that the surface energy is overestimated because of entropic contributions to the energy from the free methane in the crack.

8.9.6 Comparison with linear elastic fracture mechanics

Here, I compare the fracture properties obtained in my simulations with the expected values from linear elastic fracture mechanics. Figure 8.26 shows the global tensile stress

Table 8.3: Surface energies and critical energy release rates from this work. The critical energy release rates must be taken as upper estimates, since a finer resolution in the strain increments could reveal lower strains supporting crack propagation.

Parameters	Surface energy	Critical energy release rate
260 K, $L_Z = 12$ Å	$0.19\,\mathrm{J\,m^{-2}}$	$0.42\,\mathrm{J\,m^{-2}}$
260 K, $L_Z = 24$ Å	$0.20\,\mathrm{J\,m^{-2}}$	$0.37\,\mathrm{J\,m^{-2}}$
260 K, $L_Z = 144$ Å	$0.20\,\mathrm{J\,m^{-2}}$	$0.40\,\mathrm{J\,m^{-2}}$
280 K, $L_Z = 145$ Å	$0.21\,\mathrm{J\,m^{-2}}$	$0.38\,\mathrm{J\,m^{-2}}$

in the thick simulations plotted against the crack length at all times: Each simulation creates a trajectory on the stress–crack-length axes, corresponding to relaxation, loading and fracture of the system. By assuming that the crack is thin, the crack length can be calculated by $l_{\mathrm{crack}} \approx A_s/2L_z$. This probably slightly overestimates the crack length, since the cracks are elliptical prior to crack propagation. *Differences* between simulations should still be possible to assess with confidence. Results are presented both for simulations at $T = 260\,\mathrm{K}$ and at $T = 280\,\mathrm{K}$.

Linear elastic fracture mechanics supplies fracture criteria as relations between the elastic properties, geometry and surface energy of the crack system and the critical stress needed for that system to fail. Along with the trajectory of the simulations on the stress–crack-length axes, I have plotted three such fracture criteria: The Inglis formula, the Griffith formula, and the finite-width corrected Griffith formula:

$$\sigma_{\mathrm{Inglis}} = \sqrt{\frac{E\gamma_s}{4a}} \tag{8.12}$$

$$\sigma_{\mathrm{Griffith}} = \sqrt{\frac{2E\gamma_s}{\pi a}} \tag{8.13}$$

$$\sigma_{\mathrm{Finite\text{-}width\ Griffith}} = \sqrt{\frac{2E\gamma_s}{2W\tan\left(\frac{\pi a}{2W}\right)}} \tag{8.14}$$

These lines are plotted using the obtained values of the elastic properties from this work, and a surface energy of $0.21\,\mathrm{J\,m^{-2}}$, which corresponds to the measured surface energy of the $T = 280\,\mathrm{K}$ simulations. I have indicated in the figure where the crack evolution goes from slow crack propagation (melting) to fast crack propagation (fracture). This point is found by inspecting the crack speed plot, finding the time just before the crack tip speed starts to increase rapidly. This point in time is quite well-defined. The markers in figure 8.26 indicate the stress–crack-width configuration at this specific time for each simulation. The results agree well with the Griffith theory for brittle solids since all points line up close to the theoretical line. This figure can aid the understanding of two separate mechanisms for fracture: Some process slowly increases the crack width and reduces the tensile stress, until – at some point – the stress–crack-width configuration is such that it allows for rapid crack growth. Rapid crack growth can probably be facilitated if the slope of the stress–crack-width line has a slope such that it will cross a critical line, that is probably close to the finite-width Griffith line.

The markers corresponding to each temperature seem to form two different slopes. The slope of the markers associated with a temperature of 260 K seems slightly shifted

with respect to that corresponding to a temperature of 280 K. I speculate that this can be attributed to the pressure of the methane in the initial crack. The amplitude of the shift is compatible with a surface energy reduction by the methane of about $0.02\,\mathrm{J\,m^{-2}}$. This estimate was obtained by looking at the effect of reducing the surface energy in the finite width-corrected Griffith formula, and a reduction of $0.02\,\mathrm{J\,m^{-2}}$ shifted the Griffith-prediction by approximately the same amount as the observed shift between the lines of markers corresponding to different temperatures.

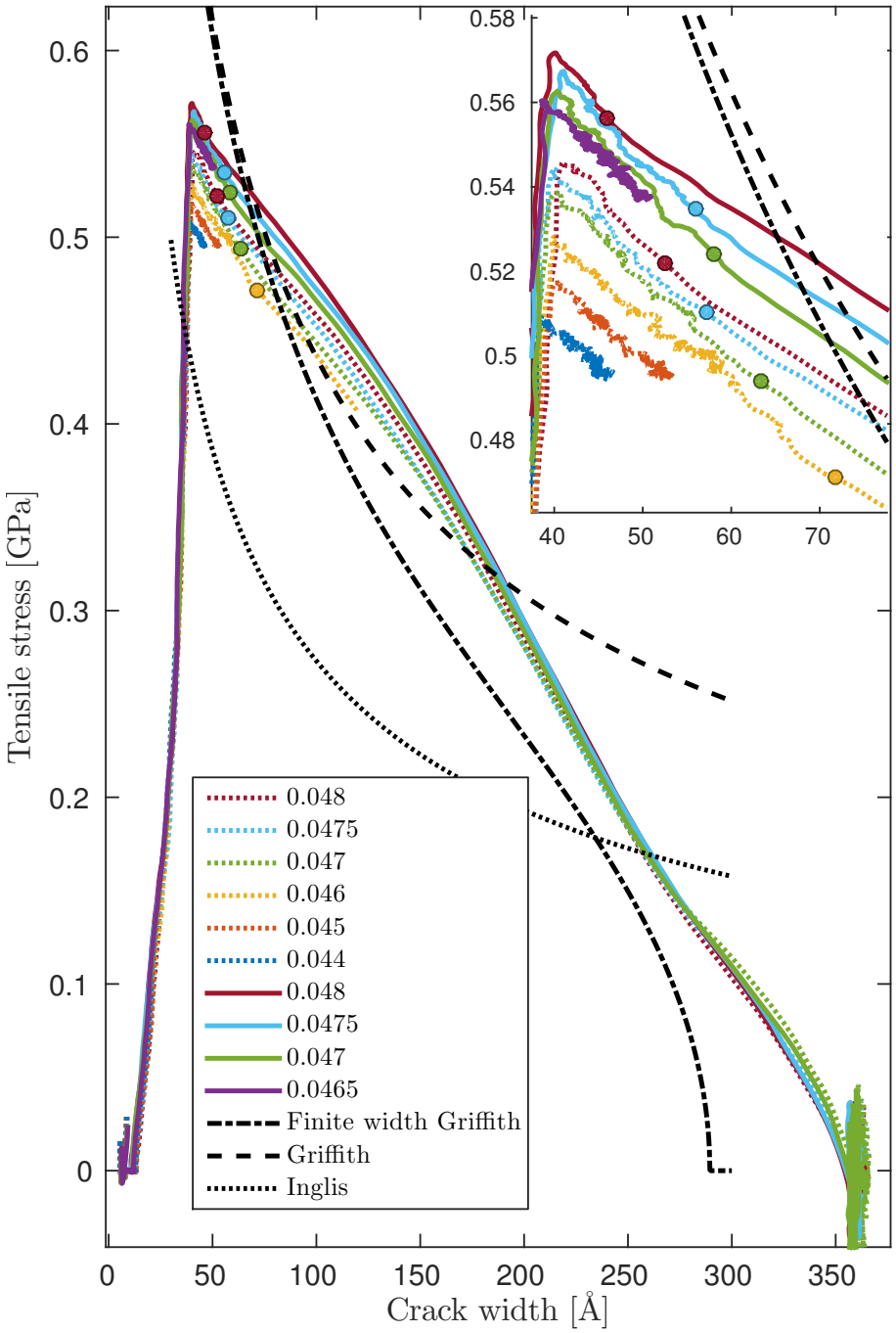


Figure 8.26: Relationship between tensile stress and crack length during dynamic runs. Markers indicate where critical fracture is estimated to have started (judgment call from looking at the fracture speed). The inset shows a zoom of the part of the figure related to the transition from slow to rapid crack growth. It is the position of the markers that determine whether this is a good agreement with Griffith or Inglis theory. The crack length is estimated as $A_s/2L_z$, and it is clear that it is over-estimated at the end of the simulation (at full crack opening). It is not clear whether the crack length is over-estimated at the early stages of crack propagation. When straining is finished, which is at the point where the stress reaches its maximum, the crack area is quite exactly estimated to the length of the crack that was carved out, namely 40 Å. The theory curves (Inglis and Griffith) are calculated using $E = 7.1\,\mathrm{GPa}$ and $\gamma_s = 0.21\,\mathrm{J\,m^{-2}}$ The finite width curve is made using equation 3.32. The legend indicates the maximum strain level in each simulation, and the different theoretical curves.

Part III

Summary and conclusions

Chapter 9

Summary and conclusions

In chapter 2, I presented the current state of the science of methane hydrates, and noticed that there have been no molecular dynamics studies on fracture of methane hydrates. I also outlined a selection of questions that I wanted to answer. The goal of this thesis has been to develop a protocol for imposing and studying fracture in methane hydrates, and to use this protocol to characterize the fracture and improve the understanding of this process.

In this chapter, I summarize my findings and the conclusions I draw from them.

9.1 Summary and conclusions

I have modeled systems of pure methane hydrates of the sI structure using LAMMPS. I have introduced artificial flaws – elliptical prismatic cracks – in the systems, and subjected them to tensile strain.

Fracture toughness and brittleness: When subjected to a stress intensity factor of ~ 0.06 MPam$^{\frac{1}{2}}$, the hydrate failed, and a crack started propagating. For the particular geometry of my systems, this happed at strains of ~ 0.05. The system was unable to deform plastically prior to failure, and when crack propagation started, the crack went all the way to the periodic boundaries of the system – the methane hydrate appeared brittle. The brittleness was further confirmed by the strain energy required to propagate a crack being almost equal to the energy associated with the crack surface.

Fracture of large systems: Even though the required strain for failure seemed quite well defined, the rupture was not immediate at this strain level. There was a waiting time from straining until rupture. It seems, but based on a limited amount of statistics, that this waiting time was more predictable for higher than for lower temperatures – at least in the sense that a lower strain gave a longer waiting time. This observation likely resulted from the slow dissociation process (melting) being more prominent at higher temperatures. Furthermore, the relation between the applied stress and the critical crack length seems to more robustly characterize failure than a waiting time after applying a certain strain: The stress–crack length curve after loading and before rapid failure had a well-defined slope, and the point where the transition from slow to fast crack propagation happened placed itself close to the theoretical prediction by LEFM. I believe that the fracture toughness and the ductility of methane hydrates

under tensile loading is highly temperature-dependent because of this effect. In further investigations, it will probably be fruitful to quantitatively characterize the relation between the stress on the hydrate in the vicinity of the crack and the dissociation rates.

Methane molecules in the fracture: When the hydrate failed, methane was almost immediately released, but only from a narrow region of the wall close to the crack pore space. Methane from about half the width of a unit cell in the system was released.

Crack instabilities: Crack instabilities have not been prominent, even though the crack speeds were measured to be within the range where crack instabilities are expected to exist. This could very well be because the modeled system was too short for instabilities to develop.

Fracture energy: When calculating the fracture energy, I found that the full potential energy of the system was a bad estimate of the free energy, because most of the energy introduced during straining was entropic energy supplied by the thermostat. The strain energy had to be calculated by explicitly integrating the stress with the strain.

Large-scale simulations: On the more practical part, I have obtained experience with performing molecular dynamics simulations. Quite early, I decided that I wanted to make a system for setting up and running simulations that was more sophisticated than editing individual input files. I ended up creating a template format and a Python code to expand the templates. The templates closely resemble runnable LAMMPS input files, but with some decoration to be able to customize the simulations. This made each simulation more reliable, since the template was already tested and refined. The templates are explained in appendix A.

9.2 Discussion

The most profound limitation in molecular dynamics is the representation of the interatomic interactions by classical potentials. Even with infinite computer power, the value of the simulations rely heavily on the quality and applicability of the potentials, and the modeling of water turns out to be extra challenging. It is impossible to get the exactly right behavior without including quantum effects. But the point of doing molecular dynamics instead of quantum methods it to look at processes whose time- and spatial scales are not available with quantum mechanics methods – we must deal with the limitations of molecular dynamics. Properties that are not easy to fit empirically, such as the behavior under high stresses and strains, can be hard to assess. Because of this profound limitation, the robustness of results of molecular dynamics simulations is clearly limited. For my specific simulations, the quantitative results are probably way off since the potentials were not specifically developed to reproduce the correct mechanical properties or the growth- and dissociation rates of methane hydrates. Indeed, while Young's modulus showed good correspondence with experiments, the Poisson's rate was far from the experimental value (0.41 in my work vs. 0.317 from experiments). However, since the potential can comfort the structure of the methane hydrate, and is able to spontaneously grow the hydrate, the qualitative properties arising from the model are probably of interest. Particularly, governing mechanisms in the failure of methane hydrates were identified: The slow dissociation (melting) is important for the

fast crack propagation. These kinds of mechanisms, involving the interplay between the external stress on a crack and the behavior of individual molecules close to the crack and the crack tip, can probably be usefully identified from molecular dynamics simulations.

Further on the limitations, the systems I modeled were very clean, in at least four ways:

- The hydrate structure was a pure sI crystal.
- There were no impurities – particles of other types than water and methane. Impurities will probably exist in an experimental or geological setting, both in form of other guest molecules and water soluble substances like salt.
- The loading condition was purely tensile, and normal to the column of unit cells, favoring a single crack direction.
- The initial crack was an elliptical prism spanning the whole z-direction. This was done for efficient comparison with linear elastic fracture mechanics, but it can also result in the system behaving almost two-dimensional.

It is possible that the brittleness I observe in my simulations can be partly attributed to these purities, and not just the short time-scale.

Molecular dynamics simulations are limited in time and space, because of limited amounts of computing power. That means some processes cannot be studied. There might, and probably do, exist processes that are too slow to be captured efficiently in MD simulations. The straining rates of my simulations were set high because they had to, not because the high straining rates were experimentally or geologically realistic. Thus, the state of the system subjected to more realistic loading rates may be different than the state of the systems I modeled. That might change the behavior during crack initiation. The brittle appearance of the methane hydrate in my simulations stands in contrast to the experimental observations mentioned in chapter 2, where methane hydrates showed a great ability to deform plastically. This may very well be a result of the high strain rate, disallowing the system to reorganize to adapt to the strain.

9.3 Outlook

From a practical point of view, the most interesting part is probably how dissociation and stability of methane hydrates can be predicted and controlled. Predictions are necessary to assess the safety of extracting methane from hydrates, and to find out whether methane hydrates can pose risks to life on earth. Controlling the dissociation is crucial to be able to produce methane from the hydrates. Reaching a level of sophistication in the description of methane hydrates where this is possible is probably closely tied to answering more fundamental scientific questions. The questions of chapter 2.8 are such questions. Based on the knowledge I have obtained during the work with this thesis, I propose topics for future work, that can both build on my work, and contribute to answering the more general questions regarding how methane hydrates can impact peoples lives:

- Further investigations to confirm the two stages of dissociation: Thermally activated crack growth (slow) and strain-activate crack growth (fast). Particularly, a study of the dissociation properties under the extreme conditions near the crack tip would be interesting to elucidate the interplay between the fast and slow fracture mechanisms.
- Other potentials. A water potential should be parametrized with a focus on reproducing the correct mechanical behavior, and at the same time have realistic growth- and dissociation rates of methane hydrates. Profound problems concerning the modeling should be taken into account, and first-principles studies of water under high stress and strain configurations can probably contribute to potentials that can be trusted for fracture simulations.
- Longer simulations of smaller systems can serve as a tool to identify processes on longer timescales. Simulations of hydrate growth had to last for microseconds, which means this timescale is important. My simulations only lasted for around a nanosecond.

I have modeled a very simple system. Further investigations should probably attack a broader set of systems:

- Less ordered structures, i.e. not replications of a unit cell.
- Changing the hydration number. Molecular dynamics simulations can shed light on how the cage occupancy influences the mechanical properties of methane hydrates.
- Defects of different geometries.
- Other loading modes. This point is in relation to one of the points below, since shear loading requires implementation of a new feature in LAMMPS.

On the more practical part, I propose some features that would be convenient to have in the simulation package:

- TIP4P water with P^3M and triclinic box. This combination is currently not supported in LAMMPS, and that disallows imposing shear stress by shearing of the simulation box (the method of Parrinello & Rahman [56]).
- LAMMPS extensions to perform more of the analysis during simulation, to be able to analyze cracks with high time resolution.

9.4 Ending remarks

Knowledge on a molecular level only become practically interesting if is is possible to scale it to macroscopic quantities it in some way or another. Purely scientifically, the role of molecular dynamics calculations on methane hydrates is probably to provide clues about what processes should be included in upscaled models – models that gain

access to longer and larger scales by ignoring more details than the lower-scale model. This is crucial when considering the important question whose answers can impact peoples lives: Can, and how can, methane hydrates be used as an energy resource? Can methane hydrates cause uncontrollable climate change, or devastating underwater landslides, and can we prevent it?

Part IV

Appendices

Appendix A

Input file templates

To be able to do many simulations of the same kind, but with either different parameters of different random seed (to collect statistics), I have defined a way to make simulation templates. I have also made a (small) Python module `template_expander` that can be used to efficiently create systematic series of simulations from a template. My workflow has been:

1. Write a LAMMPS input script and check that it runs.
2. Substitute some commands and variables with custom identifiers.
3. Apply `template_expander` to create a series of simulations.

This way of doing simulations has turned out to be efficient. I don't provide the code for `template_expander` here, but it is available upon request.

A.1 Template format

The template format is very simple. The file is written just like a regular input file, but with the exception that some properties are marked with an '@'. These are to be substituted by simulation parameters by the `template expander`. To illustrate the template, I give the file that was used for the fracture simulations in this work. The first '@'-identifier is on line 13.

```
1  units          real
2  dimension      3
3  boundary       p p p
4  atom_style     full
5  pair_style     lj/cut/tip4p/long 1 2 1 1 0.1577 10.0
6  kspace_style   pppm/tip4p 1.0e-4
7  bond_style     harmonic
8  angle_style    harmonic
9  pair_modify    mix arithmetic # Lorenz-Berthelot mixing rules
10
11
12 read_data "s1_unit_cell_tip4p_ice.data"
13 variable Nx equal @Nx
```

```
variable Ny equal @Ny
variable Nz equal @Nz
replicate ${Nx} ${Ny} ${Nz}

pair_coeff  1   1   0.21084   3.1668
pair_coeff  2   2   0.0     0.0
pair_coeff  1   2   0.0     0.0
pair_coeff  3   3   0.29391   3.73


bond_coeff    1   1000.0  0.9572
angle_coeff   1   1000.0  104.52

group water     type  1   2
group methane   type  3


variable dumpFrequency equal 300

compute methaneMsd methane msd com yes
compute waterMsd water msd com yes

compute perAtomStress all stress/atom NULL
fix fperAtomStress all ave/atom 1 ${dumpFrequency}
    ${dumpFrequency} c_perAtomStress[1] c_perAtomStress[2]
    c_perAtomStress[3] c_perAtomStress[4] c_perAtomStress[5]
    c_perAtomStress[6]

thermo 1
thermo_style custom step time etotal ke temp pe ebond eangle
    edihed eimp evdwl ecoul elong press density lx ly lz pxx pyy
    pzz pxy pxz pyz c_waterMsd[4] c_methaneMsd[4]

dump myDump all custom ${dumpFrequency} trajectory.lammpstrj id
    element x y z vx vy vz f_fperAtomStress[1]
    f_fperAtomStress[2] f_fperAtomStress[3] f_fperAtomStress[4]
    f_fperAtomStress[5] f_fperAtomStress[6]
dump_modify myDump element O H C

variable T equal @temperature
variable Tdamp equal 1000.0
variable Pdamp equal 1000.0
variable P equal 1

variable Nthermalize equal 1e5
variable Nerate equal @Nerate
variable Nproduction equal @Nproduction

variable crackWidth equal @crackWidth
variable crackLength equal @crackLength

timestep @timeStep
neigh_modify delay 0 every 1 check yes

# Cut a vertical crack in the middle of the sample
variable xmid equal "(xlo+xhi)/2.0"
```

```
variable ymid equal "(ylo+yhi)/2.0"
variable zlo1 equal "zlo"
variable zhi1 equal "zhi"

region crackHole eprism ${xmid} ${ymid} $(v_crackWidth/2.0)
    $(v_crackLength/2.0)

delete_atoms region crackHole mol yes

fix fShake  water   shake   1.0e-4  100 0   a   1   b   1
fix fxnpt all npt temp ${T} ${T} ${Tdamp} x $P $P ${Pdamp} y $P
    $P ${Pdamp} z $P $P ${Pdamp} drag 1.0
run ${Nthermalize}
write_restart restart.*

unfix fxnpt
fix fxnvt all nvt temp ${T} ${T} ${Tdamp} drag 1.0
fix fErate all deform 1 x scale @maxStrain remap x
run ${Nerate}
write_restart restart.*

unfix fErate
run ${Nproduction}
write_restart restart.*
```

A.2 Python implementation

In order to fill in for the properties that are to be varied, properties are put in a dictionary and sent to the `template_expander` module.

```
# script using template expander

from template_expander import template_expander
import os

sys_vars = {}
sys_vars['Nx'] = [24]
sys_vars['Ny'] = [24]
sys_vars['Nz'] = [12]
sys_vars['Nerate'] = ['5e4']
sys_vars['Nproduction'] = ['4e5']
sys_vars['crackLength'] = [40.0]
sys_vars['crackWidth'] = [6.0]
sys_vars['maxStrain'] = [1.047, 1.048]
sys_vars['timeStep'] = [1.0]
base_time_limit = 18.0
base_tasks = 20*15
input_folder = os.environ['TEMPLATEHOME']+'/s1_hydrate_crack'
input_file = 's1_hydrate_crack.in'
output_folder =
    os.environ['OUTPUTHOME']+'/systematic_cracks_ellipsehole_thick'

```

```
22 template_expander(input_folder, input_file, sys_vars,
       output_folder, base_time_limit, base_tasks,
       account='myAccount')
```

Appendix B

Details of using the Abel computing cluster

B.1 Compiling lammps on the Abel computing cluster

It is usually quite straightforward to install the main features of LAMMPS. However, if one is to use special features such as GPU packages or th Intel Xeon Phi, it is more complicated.

To compile LAMMPS on the Abel computer cluster, one has to load the intel compiler and mpi modules, and then follow the build instructions from the LAMMPS documentation. The program used for simulations going into this thesis, was compiled with the following command:

```
# Script to build lammps with openMP and Intel on Abel (November
    21. 2014)
module load intel
module load intelmpi.intel
make yes-user-intel
make yes-user-omp
make yes-kspace
make yes-replica
make yes-molecule
make yes-rigid
make intel_cpu
```

These commands are run from `src` in the LAMMPS install folder, which was extracted from a tarball.

B.2 Submitting jobs

When expanding templates, a corresponding job script is also generated. It typically looks like:

```
#!/bin/bash
# Job name:
#SBATCH --job-name=s1_hydrate_crack.in2015-02-12T11:17:08.271798
```

```
# Project:
#SBATCH --account=myAccound
# Wall clock limit:
#SBATCH --time='10:00:00'
#SBATCH --mem-per-cpu=4000M
# CPUs:
#SBATCH --nodes=18 --ntasks-per-node=15 --cpus-per-task=1
module purge
module load intel
module load intelmpi.intel
mpirun -np 270 lmp_intel_cpu -in s1_hydrate_crack.in
```

This job script starts LAMMPS in MPI mode with 270 mpi processes.

I have made a small python module that lets me submit a job for each subdirectory containing a file `lmp_slurm_job.sh`:

```
# File: sbatch_tree.py
import subprocess
import argparse
import os.path as op
import os

def walkfunc(arg, dirname, names):
  job_script = op.join(dirname, 'lmp_slurm_job.sh')
  if op.isfile(job_script):
    if op.isfile(op.join(dirname,'log.lammps')):
      print "Seems like simulaton is running or has been run
          since there are output files in the folder"
      print dirname
    else:
      os.chdir(dirname)
      subprocess.call(['sbatch', 'lmp_slurm_job.sh'])
      print "Submitted from", job_script

if __name__=='__main__':
  parser = argparse.ArgumentParser()
  parser.add_argument('sim_root_folder', type=str)
  args = parser.parse_args()
  op.walk(args.sim_root_folder, walkfunc, None)
```

Usage:

```
python -m sbatch_tree $PWD
```

B.3 Experienced problems when using the Abel computing cluster

When performing large-scale simulations, failures on the computing cluster can consume a lot of time. I have had two main problems when performing simulations on the Abel cluster, leading to seemingly random simulations not running or finishing correctly:

- Random node failures. Some nodes do not have the correct software settings.

Small things like how each node bind the tasks to CPU-cores can make the simulation crash.

- Slow nodes – killed jobs. If the simulation is sent to nodes that are slower than expected, the time limit set for the simulation can be too short, and the simulation gets killed.

These kinds of failures are part of the challenge of performing large-scale computations. After simulations have been submitted, it must be checked that it was started correctly. After a simulation has stopped, it must be checked whether it stopped because it was finished, or if it was killed or crahsed.

Bibliography

[1] Abascal, J L F, Sanz, E, García Fernández, R, & Vega, C. 2005. A potential model for the study of ices and amorphous water: TIP4P/Ice. *The Journal of chemical physics*, **122**(23), 234511.

[2] Abraham, Farid F., Brodbeck, Dominique, Rudge, William E., & Xu, Xiaopeng. 1997. A molecular dynamics investigation of rapid fracture mechanics. *Journal of the Mechanics and Physics of Solids*, **45**(9), 1595 – 1619.

[3] Aloulou, Faouzi. 2015. *Shale gas and tight oil are commercially produced in just four countries.*

[4] Anderson, Brian J, Tester, Jefferson W, & Trout, Bernhardt L. 2004. Accurate Potentials for Argon - Water and Methane - Water Interactions via ab Initio Methods and Their Application to Clathrate Hydrates. 18705–18715.

[5] Anderson, T. L. 2005. *Fracture Mechanics: Fundamentals and Applications, Third Edition.*

[6] Barnes, Brian C, & Sum, Amadeu K. 2013. Advances in molecular simulations of clathrate hydrates. *Current Opinion in Chemical Engineering*, **2**(2), 184–190.

[7] Benham, Peter Philip, Crawford, Roy J, & Armstrong, Cecil G. 1996. *Mechanics of engineering materials.* Longman Harlow, Essex, England.

[8] Berendsen, H J C, Postma, J P M, van Gunsteren, W F, DiNola, a, & Haak, J R. 1984. Molecular dynamics with coupling to an external bath. *The Journal of Chemical Physics*, **81**(1984), 3684–3690.

[9] Berendsen, Herman JC, Postma, James PM, van Gunsteren, Wilfred F, & Hermans, Jan. 1981. Interaction models for water in relation to protein hydration. *Pages 331–342 of: Intermolecular forces.* Springer.

[10] Bernal, JD, & Fowler, RH. 1933. A theory of water and ionic solution, with particular reference to hydrogen and hydroxyl ions. *Journal of Chemical Physics*, **1**(8), 515–548.

[11] Bhattacharya, Supriyo, Coasne, Benoit, Hung, Francisco R., & Gubbins, Keith E. 2009. Modeling micelle-templated mesoporous material SBA-15: Atomistic model and gas adsorption studies. *Langmuir*, **25**(10), 5802–5813.

[12] Bouchbinder, Eran, Mathiesen, Joachim, & Procaccia, Itamar. 2004. Roughening of fracture surfaces: The role of plastic deformation. *Physical Review Letters*, **92**(24), 245505–1.

[13] Bücker, Christian, Jenisch, Uwe, Lutter, Stephan, Matz-Lück, Nele, Messner, Jürgen, Petersen, Sven (Leibniz Institute of Marine Science), Rüpke, Lars H., Schwarz-Schampera, Ulrich, & Wallmann, Klaus. 2014. Marine Resources - Opportunities and Risks. *World Ocean Review*, **3**, 165.

[14] Buehler, M. J. 2008. *Atomistic Modeling of Mechanical Failure*.

[15] Buehler, Markus J, & Gao, Huajian. *Modeling Dynamic Fracture Using Large-Scale Atomistic Simulations*. Chap. 1, pages 1–68.

[16] Cao, Dapeng, & Wu, Jianzhong. 2004. Self-diffusion of methane in single-walled carbon nanotubes at sub- and supercritical conditions. *Langmuir : the ACS journal of surfaces and colloids*, **20**(9), 3759–3765.

[17] Central Intellegence Agency. 2013. *The World Factbook 2013-14*.

[18] Circone, Susan, Kirby, Stephen H., & Stern, Laura a. 2005. Direct measurement of methane hydrate composition along the hydrate equilibrium boundary. *Journal of Physical Chemistry B*, **109**(3), 9468–9475.

[19] Cornell, Wendy D, Cieplak, Piotr, Gould, Ian R, Merz, Kenneth M, Ferguson, David M, Spellmeyer, David C, Fox, Thomas, Caldwell, James W, & Kollman, Peter a. 1995. A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules. *Society*, 5179–5197.

[20] Daw, Murray S., & Baskes, M. I. 1984. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Phys. Rev. B*, **29**(Jun), 6443–6453.

[21] Deserno, Markus, & Holm, Christian. 1998. How to mesh up Ewald sums. I. a theoretical and numerical comparison of various particle mesh routines. *Journal of Chemical Physics*, **109**(1998), 7678–7693.

[22] Durham, William B. 2003. The strength and rheology of methane clathrate hydrate. *Journal of Geophysical Research*, **108**(June), 1–11.

[23] Ewald, P P. 1921. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Annalen der Physik*, **369**(3), 253–287.

[24] Finney, John L. 2004. Water? What's so special about it? *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, **359**(1448), 1145–1163; discussion 1163–1165, 1323–1328.

[25] Fock, V. 1930. Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems. *Zeitschrift für Physik*, **61**(1-2), 126–148.

[26] Frenkel, Daan, & Smit, Berend. 2001. *Understanding molecular simulation: from algorithms to applications*. Vol. 1. Academic press.

[27] González, Miguel Angel, & Abascal, Joś L F. 2010. The shear viscosity of rigid water models. *Journal of Chemical Physics*, **132**, 1–2.

[28] Griffith, Alan A. 1920. The phenomena of rupture and flow in solids. *Philosophical transactions of the royal society of london. Series A, containing papers of a mathematical or physical character*, 163–198.

[29] Hammerschmidt, E G. 1934. Formation of gas hydrates in natural gas transmission lines. *J. Ind. Eng. Chem. (Washington, D. C.)*, **26**(4), 851–855.

[30] Hantal, György, Brochard, Laurent, Laubie, Hadrien, Ebrahimi, Davoud, Pellenq, Roland J.-M., Ulm, Franz-Josef, & Coasne, Benoit. 2014. Atomic-scale modelling of elastic and failure properties of clays. *Molecular Physics*, **112**(January 2015), 1294–1305.

[31] Hester, Keith C, & Brewer, Peter G. 2009. Clathrate hydrates in nature. *Annual review of marine science*, **1**(1), 303–327.

[32] Hockney, R. W., & Eastwood, J. W. 1988. *Computer Simulation Using Particles*. Bristol, PA, USA: Taylor & Francis, Inc.

[33] Holland, Dominic, & Marder, M. 1998. Ideal Brittle Fracture of Silicon Studied with Molecular Dynamics. *Phys. Rev. Lett.*, **80**(Jan), 746–749.

[34] Hoover, William G. 1985. Canonical dynamics: Equilibrium phase-space distributions. *Physical Review A*, **31**(3), 1695–1697.

[35] Humphrey, William, Dalke, Andrew, & Schulten, Klaus. 1996. VMD – Visual Molecular Dynamics. *Journal of Molecular Graphics*, **14**, 33–38.

[36] Hünenberger, Philippe H. 2005. Advanced Computer Simulation. **173**, 105–149.

[37] Inglis, Charles Edward. 1913. Stresses in a plate due to the presence of cracks and sharp corners. *Transactions of the Institute of Naval Architects*, **55**, 219–241.

[38] Irwin, George Rankine. 1956. Onset of Fast Crack Propagation in High Strength Steel and Aluminum Alloys. *Pages 289–305 of: Sagamore Research Conference Proceedings*, vol. 2.

[39] Jacobson, Liam C, & Molinero, Valeria. 2010. A methane-water model for coarse-grained simulations of solutions and clathrate hydrates. *The journal of physical chemistry. B*, **114**(21), 7302–11.

[40] Jorgensen, William L, & Tirado-Rives, Julian. 1988. The OPLS Potential Functions for Proteins. Energy Minimizations for Crystals of Cyclic Peptides and Crambin. *Journal of the American Chemical Society*, **110**(4), 1657–1666.

[41] Jorgensen, William L., Chandrasekhar, Jayaraman, Madura, Jeffry D., Impey, Roger W., & Klein, Michael L. 1983. Comparison of simple potential functions for simulating liquid water. *The Journal of Chemical Physics*, **79**(2), 926.

[42] Kennett, James P. 2003. *Methane hydrates in Quaternary climate change: The clathrate gun hypothesis*. American Geophysical Union.

[43] Laubie, Hadrien, & Ulm, Franz-Josef. 2014. Plane-Strain Crack Problem in Transversely Isotropic Solids for Hydraulic Fracturing Applications. *Journal of Engineering Mechanics*, 1–12.

[44] Mahoney, Michael W., & Jorgensen, William L. 2000. A five-site model for liquid water and the reproduction of the density anomaly by rigid, nonpolarizable potential functions Michael. *Journal of Chemical Physics*, **112**(20).

[45] Makogon, Y.F., Holditch, S.A., & Makogon, T.Y. 2007. Natural gas-hydrates — A potential energy source for the 21st Century. *Journal of Petroleum Science and Engineering*, **56**(1–3), 14 – 31. Natural Gas Hydrate / Clathrate The Major Organic Carbon Reserve of the Earth.

[46] Malischewsky, Peter G. 2005. *Nanotechnology*, **16**(6), 995.

[47] Martin, Marcus G, & Siepmann, J Ilja. 1998. Transferable Potentials for Phase Equilibria. 1. United-Atom Description of n -Alkanes. **5647**(97), 2569–2577.

[48] Martyna, Glenn J., Tobias, Douglas J., & Klein, Michael L. 1994. Constant pressure molecular dynamics algorithms. *The Journal of Chemical Physics*, **101**(5), 4177.

[49] Matsumoto, Masakazu, Saito, Shinji, & Ohmine, Iwao. 2002. Molecular dynamics simulation of the ice nucleation and growth process leading to water freezing. *Nature*, **416**(6879), 409–13.

[50] Milkov, Alexei V. 2004. Global estimates of hydrate-bound gas in marine sediments: how much is really out there? *Earth-Science Reviews*, **66**(3–4), 183 – 197.

[51] Miyamoto, Shuichi, & Kollman, Peter a. 1992. Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models. *Journal of Computational Chemistry*, **13**(8), 952–962.

[52] Ning, Fulong, Glavatskiy, Kirill, Vlugt, Thijs, & Kjelstrup, Signe. 2010. Lattice parameters and corresponding properties of methane and carbon dioxide hydrates: molecular dynamics simulations. **c**(1).

[53] Ning, Fulong, Yu, Yibing, Kjelstrup, Signe, Vlugt, Thijs J. H., & Glavatskiy, Kirill. 2012. Mechanical properties of clathrate hydrates: status and perspectives. *Energy & Environmental Science*, **5**(5), 6779.

[54] Nosé, Shuichi. 1984. A unified formulation of the constant temperature molecular dynamics methods. *The Journal of Chemical Physics*, **81**(1984), 511–519.

[55] Pachauri, Rajendra K, Meyer, Leo, Van Ypersele, Jean-Pascal, Brinkman, Sander, Van Kesteren, Line, Leprince-Ringuet, Noëmie, & Van Boxmeer, Fijke. 2014. *Climate Change 2014 Synthesis Report. Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. Tech. rept. Geneva, Switzerland.

[56] Parrinello, M., & Rahman, a. 1981. Polymorphic Transitions in Single Crystals: a New Molecular Dynamics Method. *Journal of Applied Physics*, **52**(12), 7182–7190.

[57] Plimpton, Steve. 1995. Fast Parallel Algorithms for Short – Range Molecular Dynamics. *Journal of Computational Physics*, **117**(June 1994), 1–19.

[58] Pozhar, Liudmila. 1995. *Transport theory of inhomogeneous fluids: fundamentals and applications*. World Scientific Series in Contemporary Chemical Physics. Singapore: World Scientific.

[59] Quesnel, D. J., Rimai, D. S., & DeMejo, L. P. 1993. Elastic compliances and stiffnesses of the fcc Lennard-Jones solid. *Phys. Rev. B*, **48**(10), 6795–6807.

[60] Ryckaert, Jean-Paul, Ciccotti, Giovanni, & Berendsen, Herman J.C. 1977. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *Journal of Computational Physics*, **23**, 327–341.

[61] sandia.gov. 2015 (April). *LAMMPS Documentation*. http://lammps.sandia.gov/doc/Manual.html.

[62] Shinoda, Wataru, Shiga, Motoyuki, & Mikami, Masuhiro. 2004. Rapid estimation of elastic constants by molecular dynamics simulation under constant stress. *Physical Review B*, **69**(13), 134103.

[63] Slater, John C. 1930. Note on Hartree's method. *Physical Review*, **35**(2), 210.

[64] Stern, Laura a, Kirby, Stephen H, & Durham, William B. 1998. Polycrystalline Methane Hydrate : Synthesis from Superheated Ice, and Low-Temperature Mechanical Properties. *Energy and Fuels*, **12**(97), 201–211.

[65] Stillinger, Frank H., & Weber, Thomas a. 1985. Computer simulation of local order in condensed phases of silicon. *Physical Review B*, **31**(8), 5262–5271.

[66] Subramanian, S, Kini, R A, Dec, S F, & Jr, E D Sloan. 2000. Evidence of structure {II} hydrate formation from methane+ethane mixtures. *Chemical Engineering Science*, **55**(11), 1981–1999.

[67] Sultan, N., Cochonat, P., Foucher, J.-P., & Mienert, J. 2004. Effect of gas hydrates melting on seafloor slope instability. *Marine Geology*, **213**(1–4), 379 – 401. {COSTA} - Continental Slope Stability.

[68] Takeuchi, Fumihito, Hiratsuka, Masaki, Ohmura, Ryo, Alavi, Saman, Sum, Amadeu K, & Yasuoka, Kenji. 2013. Water proton configurations in structures I, II, and H clathrate hydrate unit cells. *The Journal of chemical physics*, **138**(12), 124504.

[69] Tazi, Sami, Boţan, Alexandru, Salanne, Mathieu, Marry, Virginie, Turq, Pierre, & Rotenberg, Benjamin. 2012. Diffusion coefficient and shear viscosity of rigid water models. *Journal of physics. Condensed matter : an Institute of Physics journal*, **24**(28), 284117.

[70] The Norwegian National Committees for Research Ethics. *Regarding the assessment of research ethics in petroleum research.*

[71] Thompson, Aidan P., Plimpton, Steven J., & Mattson, William. 2009. General formulation of pressure and stress tensor for arbitrary many-body interaction potentials under periodic boundary conditions. *Journal of Chemical Physics*, **131**(2009).

[72] Vega, Carlos, & Abascal, Jose L. F. 2011. Simulating water with rigid non-polarizable models: a general perspective. *Physical Chemistry Chemical Physics*, **13**(44), 19663.

[73] Voigt, W. 1928. Lehrbuch der Kristallphysik. *Teubner, Leipzig.*

[74] Waite, W F, Helgerud, M B, Nur, a, Pinkston, J C, Stern, L a, Kirby, S H, & Durham, W B. 2000. Laboratory measurements of compressional and shear wave speeds through methane hydrate. *Gas Hydrates: Challenges For The Future*, **912**, 1003–1010 ST – Laboratory measurements of compres.

[75] Wallmann, Klaus, Pinero, Elena, Burwicz, Ewa, Haeckel, Matthias, Hensen, Christian, Dale, Andrew, & Ruepke, Lars. 2012. The global inventory of methane hydrate in marine sediments: A theoretical approach. *Energies*, **5**(7), 2449–2498.

[76] Walsh, Matthew R, Koh, Carolyn a, Sloan, E Dendy, Sum, Amadeu K, & Wu, David T. 2009. Microsecond simulations of spontaneous methane hydrate nucleation and growth. *Science (New York, N.Y.)*, **326**(5956), 1095–8.

[77] Xu, Wenyue, & Germanovich, Leonid N. 2006. Excess pore pressure resulting from methane hydrate dissociation in marine sediments: A theoretical approach. *Journal of Geophysical Research*, **111**(B1), B01104.

[78] Yeh, In Chul, & Hummer, Gerhard. 2004. System-size dependence of diffusion coefficients and viscosities from molecular dynamics simulations with periodic boundary conditions. *Journal of Physical Chemistry B*, **108**, 15873–15879.

[79] Zhou, S. J., Lomdahl, P. S., Thomson, R., & Holian, B. L. 1996. Dynamic Crack Processes via Molecular Dynamics. *Phys. Rev. Lett.*, **76**(Mar), 2318–2321.